

IL PRIMO SETTIMANALE DI SOFTWARE SU CARTA

L. 1000

PER **Apple - TI99/4A - MSX**

Una pubblicazione della J.soft editrice

sped. in abb. post. Gruppo 1/70



PAPER SOFT

Anno 2 - N° 16 - 26 aprile 1985



apple

**Documenta-
zione
delle shape
table
Torna
indietro**



apple

**Ochello
Dischi
personaliz-
zati**

**Disassembler
Tank**



TI-99/4A

**Solitario
Paratrooper**



TI-99/4A

MSX

**Tiro
al bersaglio**

MSX

**Robot
Corsa
campestre**

IN EDICOLA

SUPER

sinc

SUPER

COMMODORE

N°4



**SuperSinc e
Super Commodore
sono idee**

J.soft EDITRICE

**CON
CASSETTA**



MSX

- 4** Documentazione delle shape table
di E. Ringot trad. e adatt. di M. Cerofolini
- 6** Torna indietro (Paddies)
di K. Stinger trad. e adatt. di M. Cerofolini
- 7** Othello
di R. O. Duba trad. e adatt. di M. Cerofolini
- 10** Dischi personalizzati
di P. Bouteux trad. e adatt. di M. Cerofolini

- 22** Corsa campestre
di W. Snijers trad. e adatt. di P. Maffei
- 25** Robot
di W. Snijers trad. e adatt. di P. Maffei
- 27** Tiro al bersaglio
di W. Snijers trad. e adatt. di P. Maffei



- 12** Disassembler
di J. Dunn trad. e adatt. di E. Re Garbagnati
- 14** Tank
di A. Loes trad. e adatt. di E. Re Garbagnati
- 17** Solitario
di W. Lenk trad. e adatt. di E. Re Garbagnati
- 20** Paratrooper (Extended BASIC)
di J. Goetz trad. e adatt. di E. Re Garbagnati

lsoft s.r.l.

DIREZIONE, REDAZIONE, AMMINISTRAZIONE

V.le Restelli, 5
20124 MILANO

Tel. (02) 68.88.228-68.37.97

DIRETTORE RESPONSABILE:

Pietro Dell'Orco

COORDINAMENTO TECNICO:

Mauro Cristobal Grizzi

REDAZIONE:

Paolo Maffei
Carlo Panzani
Emilio Re Garbagnati

GRAFICA E IMPAGINAZIONE:

Margherita La Noce
Ivana Rossi
Raffaella Toffolatti

FOTOCOPOSIZIONE:

d5b Via Vignola, 5
Tel. 02/59.65.08
20133 MILANO

CONTABILITÀ:

Giulia Pedrazzoni
Flavia Bonatti

AUTORIZZAZIONE ALLA PUBBLICAZIONE:

Tribunale di Milano n° 200
del 14/04/1984

STAMPA:

Intergrafica - Pioltello (MI)
Rivista associata
all'Unione stampa
Periodica Italiana



PUBBLICITÀ

Concessionaria per l'Italia e l'Estero
L'Advertising s.r.l.

V.le Restelli, 5
20124 MILANO

Tel. (02) 68.82.895-68.80.606-68.87.233

Tlx. 316213 REINA I

Concessionaria esclusiva per la
DIFFUSIONE in Italia e Estero:

SODIP - Via Zuretti, 25
20125 MILANO

Spedizione in abbonamento

postale Gruppo II/70

Prezzo della rivista L. 1.000

Numeri arretrati L. 2.000

© TUTTI I DIRITTI DI
RIPRODUZIONE O TRADUZIONE
DEGLI ARTICOLI PUBBLICATI
SONO RISERVATI

Documentazione delle shape table



Questo programma permette di documentare le shape table (tabelle di forme) ed in particolare consente di avere:

- La lunghezza della tabella.
- Il numero delle shape (forme).
- Per ciascuna forma l'indirizzo relativo in rapporto all'inizio della tabella.

Si possono inoltre vedere le forme su schermo o su stampante.

Le tabelle vengono caricate all'indirizzo \$4000 (decimale 16384) sotto la pagina ad alta risoluzione HGR2. Il programma di stampa funziona solo con la stampante SEIKOSHA GP.100A.

```
100 REM -----
110 REM DOCUMENTATION
120 REM DELLE SHAPE
130 REM TABLE
140 REM -----
```

```
150 REM *****OOC/SHAPE*****
```

```
160 REM
```

```
-1-INIZIALIZZAZIONE
```

```
170 LOMEM: 24576: REM DOPO HGR2
```

```
180 ONERR GOTO 1150
```

```
190 TEXT: HOME
```

```
200 PRINT "NOME DELLA TABELLA..."
```

```
210 INPUT "N°: IF N° = "" THEN  
ENO
```

```
220 GOTO 1010
```

```
230 REM
```

```
-2-MESSAGGIO
```

```
240 VTAO 20: INVERSE: PRINT MESSAGE$: FOR I = 1 TO 1000: NORMAL  
: NEXT: VTAB 20: CALL -B6  
B: RETURN
```

```
250 REM
```

```
-3-VERIFICA TABELLA
```

```
260 A = 4 * 16 * 3: POKE 232,0: POKE  
233,64: REM INDIRIZZO TABELLA
```

```
270 MESSAGE$ = "CARICAMENTO DI " +  
N$: GOSUB 240
```

```
280 PRINT CHR$(4)"BLOAD"N$,A"
```

```
290 N = PEEK(A): REM NUMERO DI  
I SHAPES
```

```
300 DIM ADRESSE(N): REM TABELLA  
DEGLI INDIRIZZI RELATIVI
```

```
310 FOR I = 1 TO N
```

```
320 AL = PEEK(A + 2 * I): REM  
INDIRIZZO RELATIVO OTTETTO
```

```
330 AH = PEEK(A + 2 * I + 1): REM  
INDIRIZZO RELATIVO OTTETTO
```

```
340 ADRESSE(I) = AH * 256 + AL: IF  
A + ADRESSE(I) > 2 * 16 THEN  
1210: REM INDIRIZZO RELATIVO
```

```
350 ZERO = 0: IF I > 1 THEN ZERO =  
PEEK(A + ADRESSE(I) - 1): REM  
ULTIMO OTTETTO DELLA SHAPE  
PRECEDENTE
```

```
360 IF ZERO < 0 THEN GOTO 12  
10: REM NON E' UNA SHAPE TABLE!!!
```

```
370 NEXT I
```

```
380 MESSAGE$ = N$ + " E' UNA SHAPE  
TABLE": GOSUB 240
```

```
390 RETURN
```

```
400 REM
```

```
-4-LUNGHEZZA DELLA TABELLA
```

```
410 D = A + ADRESSE(N): REM INDIRIZZO  
ULTIMA SHAPE
```

```
420 IF PEEK(D) < 0 THEN D =  
0 + 1: GOTO 420: REM LA TABELLA  
TERMINA CON UNO ZERO
```

```
430 LL = D - A + 1: REM LUNGHEZZA  
A DECIMALE
```

```
440 REM CONVERSIONE ESADECIMALE
```

```
450 MSB% = LL / 256: LSB% = LL - 256 * MSB%
```

```
460 O%(1) = MSB% / 16: O%(2) = MSB% /  
16: O%(4) = LSB% - 16 * O%(3):  
: REM QUADRUPE COSTITUENTI  
L'INDIRIZZO
```

```

470 FOR I = 1 TO 4:O%(I) = STR%
  (O%(I))
480 IF O%(I) > 9 THEN O%(I) = CHR%
  (O%(I) + 55)
490 NEXT
500 RETURN
510 REM

```

-5-RAPPRESENTAZIONE

```

520 IF IR% = 1 THEN POKE -162
  97,0: POKE -16302,0: POKE
  -16304,0: GOTO 710
530 HGR : POKE -16302,0
540 IR% = 1
550 HCOLOR= 3: SCALE= 1: ROT= 0
560 O = INT ( SQR (N)): REM NB.L
  IGNES
570 PY = INT (192 / O): REM INTE
  RVALLE EN Y
580 MY = (192 - (O - 1) * PY) / 2
  : REM MARGE EN Y
590 P = INT (N / O): IF N > P *
  O THEN P = P + 1
600 PX = INT (280 / P)
610 MX = (280 - (P - 1) * PX) / 2

620 FOR L = 1 TO O
630 FOR F = 1 TO P
640 I = F + (L - 1) * P
650 IF I > N THEN 710
660 Y = MY + (L - 1) * PY: REM C
  QTE
670 X = MX + (F - 1) * PX: REM AB
  SCISSE
680 DRAW 1 AT X,Y
690 NEXT F
700 NEXT L
710 GET A$: TEXT
720 RETURN
730 REM

```

-6-SU STAMPANTE

```

740 REM
(SOLO STAMPANTE SEIKOSH
  A GP100A SLOT1)

```

```

750 PRINT CHR% (13) CHR% (4)"PR
  #1"
760 PRINT CHR% (20)
770 PRINT CHR% (14) TAB( 20 - LEN
  (N%) / 2)N% CHR% (15): PRINT

780 GOSUB 840
790 PRINT
800 POKE -12317,0
810 PRINT CHR% (17)
820 PRINT CHR% (4)"PR#0"
830 RETURN
840 REM

```

-7-TABELLA INOIRIZZ1

```

850 TEXT : HOME : IEX = 1
860 PRINT "LUNGHEZZA"

```

```

870 PRINT "DECIMALE" -> "LL
880 PRINT "ESADECIMALE" -> "%D%
  (1)0%(2)0%(3)0%(4)
890 PRINT : PRINT "NUMERO DI FOR
  ME" "N
900 UTAB 9
910 PRINT "FORMA": TAB( 15);"IND
  IRIZZ0"
920 POKE 34,10: UTAB 11
930 FOR I = 1 TO N
940 PRINT TAB( 5 - LEN ( STR%
  (I));I; TAB( 20 - LEN ( STR%
  (AD(I))) );ADRESSE(I)
950 IF IEX = 0 THEN 970
960 IF I = N OR I = 10 * IIT (I
  / 10) THEN GET A$: HOME
970 NEXT
980 IEX = 0: TEXT
990 RETURN
1000 REM

```

-8-MENU'

```

1010 GOSUB 240
1020 GOSUB 410
1030 HOME
1040 PRINT "COMMAND1": PRINT
1050 PRINT "DISEGNO DI SHAPES...
  ...1"
1060 PRINT "TABELLA INOIRIZZ1...
  ...2"
1070 PRINT "STAMPA.....
  ...3"
1080 PRINT "ALTRE TABELLE.....
  ...4"
1090 PRINT "FINE.....
  ...5"
1100 GET A$:AZ = VAL (A%)
1110 ON AZ GOSUB 520,850,750,112
  0,1130: GOTO 1030
1120 RUN
1130 TEXT : HOME : END
1140 REM

```

-8-TRATTAMENTO ERRORI

```

1150 ER = PEEK (222):LI = PEEK
  (218) + PEEK (219) * 256
1160 IF ER = 6 THEN MESSAGE$ = N
  % + " SCONOSCIUTO": GOSUB 24
  0: RUN
1170 TEXT
1180 PRINT CHR% (13) CHR% (4)"P
  R#0"
1190 HOME : PRINT "ERRORE TIPO "
  ER: PRINT "LINEA" "LI: END
1200 REM

```

-9-FILE NON SHAPE T.

```

1210 MESSAGE$ = N% + " NON E' UNA
  SHAPE TABLE": GOSUB 240: RUN

```

Torna indietro

Ecco un gioco che metterà a dura prova i vostri riflessi e la vostra abilità nel controllare le paddles. Si gioca in due e ciascun giocatore deve fare passare il proprio contrassegno, posto all'inizio del gioco nella parte bassa dello schermo, tra i fori che appaiono nelle pareti che via via si avvicinano. Se si riesce a centrare il foro di passaggio si procede verso l'alto dove il gioco diventa sempre più difficile perché si dispone di un tempo minore per tentare di effettuare il passaggio. Vince il giocatore che per primo riesce a raggiungere la parte più alta dello

schermo. Attenzione però che se mancate una uscita... tornate indietro al punto di partenza!! Il gioco è scritto in Integer BASIC che è una variante molto veloce del BASIC classico. Questo BASIC è posseduto da tutti gli appleisti perché fa parte dei programmi forniti con il sistema operativo DOS 3.3 o con il nuovo PRODOS. Questo linguaggio possiede una bellissima caratteristica: il controllo della sintassi al momento della introduzione delle linee del programma evitando così i classici SYNTAX ERROR al momento dell'esecuzione

```
10 TEXT : CALL -93:
20 GOSUB 450
30 GOSUB 610
40 GR
50 COLOR=7
60 FOR A=0 TO 39
70 HLINE 0,39 AT A
80 NEXT A
90 COLOR=9
100 VLINE 0,39 AT 19
110 HLINE 0,39 AT 0: HLINE 0,39 AT
39: VLINE 0,39 AT 0: VLINE 0,
39 AT 39
120 S=30: I=1: W=20: J=0: Y=3: F=1: D=
20: T=30: X=3: G=23: H=23
130 IF S<=3 THEN 490
140 IF T<=3 THEN 540
150 COLOR=0
160 B=(PDL(0)/15)+1
170 C=(PDL(1)/13)+20: IF C<=38
THEN C=38
180 PLOT B,S: PLOT C,T
190 COLOR=7: GOSUB 400
```

```
200 C=2+1: W=W+1: F=F+1: D=D+1
210 COLOR=15
220 HLINE 1,X AT Z: HLINE X+2,18 AT
Z
230 HLINE 20,G AT F: HLINE G+2,38
AT F
240 HLINE 20,H AT D: HLINE H+2,38
AT D
250 HLINE 1,Y AT W: HLINE Y+2,18 AT
W
260 IF SCRNB(B,S)=15 THEN GOSUB
470: IF SCRNB(C,T)=1 THEN S=
36
```

```
270 IF SCRNB(C,T)=15 THEN GOSUB
480: IF SCRNB(C,T)=1 THEN T=
36
280 GOSUB 400
290 COLOR=7
300 HLINE 1,Y AT W: HLINE Y+2,18 AT
W
310 HLINE 20,G AT F: HLINE G+2,38
AT F
320 HLINE 20,H AT D: HLINE H+2,38
AT D
330 HLINE 1,X AT Z: HLINE X+2,18 AT
Z
340 PLOT B,S: PLOT C,T
350 IF Z=38 THEN Z=1: IF F=38 THEN
F=1
360 IF W=38 THEN W=1: IF D=38 THEN
D=1
370 IF F=1 THEN GOSUB 430: IF D=
1 THEN GOSUB 440
380 IF Z=1 THEN GOSUB 410: IF W=
1 THEN GOSUB 420
390 GOTO 130
400 POKE 0,100: POKE 1,5: CALL
2: RETURN
410 X= RND (16)+1: S=S-3: RETURN
420 Y= RND (16)+1: S=S-3: RETURN
430 G= RND (17)+20: T=T-3: RETURN
440 H= RND (17)+20: T=T-3: RETURN
450 POKE 2,173: POKE 3,48: POKE
4,192: POKE 5,136: POKE 6,208
: POKE 7,4: POKE 8,198: POKE
9,1: POKE 10,240
```

```

400 POKE 11,8: POKE 12,202: POKE
13,208: POKE 14,246: POKE 15
16,284: POKE 16,0: POKE 17,76
: POKE 18,2: POKE 19,0: POKE
20,96: RETURN
470 POKE 0,50: POKE 1,100: COLOR=
1: PLOT 8,5: CALL 2: RETURN

480 POKE 0,100: POKE 1,100: COLOR=
1: PLOT 8,5: CALL 2: RETURN

490 COLOR=13: FOR I=1 TO 38: HLINE
1,18 AT 1: NEXT I: IF S=T THEN
590:
500 COLOR=15: VLINE 4,10 AT 7: PLOT
9,10: VLINE 4,10 AT 9: PLOT
10,10: VLINE 4,10 AT 11
510 VLINE 12,18 AT 9
520 VLINE 20,26 AT 7: HLINE 8,10 AT
21: VLINE 21,26 AT 11
530 GOSUB 710: GOTO 720
540 COLOR=14: FOR I=1 TO 38: HLINE
20,38 AT 1: NEXT I: IF S=T THEN
600
550 COLOR=15: VLINE 4,10 AT 27: PLOT
28,10: VLINE 4,10 AT 29: PLOT
30,10: VLINE 4,10 AT 31
560 VLINE 12,18 AT 29
570 VLINE 20,26 AT 27: HLINE 28,30
AT 21: VLINE 21,26 AT 31
580 GOSUB 710: GOTO 720
590 COLOR=14: FOR I=1 TO 38: HLINE
20,38 AT 1: NEXT I
591 COLOR=15: HLINE 17,21 AT 5: VLINE
5,10 AT 19
592 VLINE 12,18 AT 19
593 VLINE 20,26 AT 18: HLINE 19,21
AT 20: HLINE 19,20 AT 22: HLINE
19,21 AT 26

594 GOSUB 710: GOTO 720
600 COLOR=13: FOR I=1 TO 38: HLINE
1,18 AT 1: NEXT I
601 COLOR=15: HLINE 17,21 AT 5: VLINE
5,10 AT 19
602 VLINE 12,18 AT 19
603 VLINE 20,26 AT 18: HLINE 19,21
AT 20: HLINE 19,20 AT 22: HLINE
19,21 AT 26
604 GOSUB 710: GOTO 720
610 FOR Q=17 TO 11 STEP -1
620 UNTAB 10: TAB (0): PRINT "****TORN
A INDIETRO***": GOSUB
710: NEXT Q: FOR P=1 TO 1000
: NEXT P
630 TAB 5
640 UNTAB 15: INPUT "VUOI LE ISTRUZIONI
(S/N)?",C$
650 IF C$="N" THEN RETURN
660 CALL -936: UNTAB 10: PRINT "FAI P
ASSARE IL TUO PUNTO NEI FORI"
: PRINT "DEGLI OSTACOLI."

670 PRINT "LE PADDLES CONTROLLANO IL
MOVIMENTO DEI": PRINT "DUE GIOC
ATORI"
680 PRINT : PRINT "IL PUNTO SALE SU'
MA SE COLPISCI UN OSTACOLO.
... TORNI INDIETRO!!"
690 PRINT : INPUT "BATTI RETURN PER
INIZIARE",A$
700 RETURN
710 POKE 0,100: POKE 1,100: CALL
2: RETURN
720 PRINT : INPUT "ANCORA (S/N)
)?",A$
725 CALL -936
730 IF A$="S" THEN 40
740 END

```

Othello



Il gioco dell'Othello comincia ormai ad essere abbastanza diffuso e questa versione sfrutta la grafica del vostro Apple per renderlo più suggestivo e divertente.

Le regole del gioco sono facili ma la strategia del gioco può essere abbastanza complicata. Si gioca su una scacchiera tradizionale e ciascun giocatore inizia con una dotazione di 32 pezzi. Ogni pezzo ha una facciata bianca e l'altra nera e ciascun colore contraddistingue un giocatore. Si gioca a turno e, se un giacato-

re si trova nell'impossibilità di fare una mossa legale, perde il turno. Quando entrambi i giocatori si trovano in questa condizione la partita ha termine.

Il gioco consiste nell'occupare con il proprio pezzo un quadrato vuoto della scacchiera. Le prime due mosse di ogni giocatore devono essere fatte collocando il pezzo in uno dei quattro quadrati centrali. In seguito ciascun concorrente metterà uno dei propri pezzi su un qualsiasi quadrato vuoto della scacchiera cercando di

formare una linea retta (orizzontale, verticale, diagonale), avente ad ogni estremo uno dei propri pezzi e, nel mezzo, uno o più pezzi dell'avversario. A questo punto i pezzi dell'avversario che si trovano all'interno della linea retta vengono rovesciati e mostrano quindi le facciate col colore che contraddistingue l'altro giocatore. Vince il concorrente che, al termine del gioco, ha il maggior numero di pezzi col proprio colore.

Il gioco è scritto in Integer BASIC che è una va-

riante molto veloce del BASIC classico. Questo BASIC è posseduto da tutti gli appleisti perché fa parte dei programmi forniti con il sistema operativo DOS 3.3 o con il nuovo PRODOS. Questo linguaggio possiede una bellissima caratteristica: il controllo della sintassi al momento della introduzione delle linee del programma evitando così i classici SYNTAX ERROR al momento dell'esecuzione. Buon divertimento!

```

10 TEXT : CALL -936: POKE -16298
  ,0
30 REM GIOCO IL GIOCO DELL'OTTELLO
  CON DUE STRATEGIE:
40 REM 1. CATTURA IL MAGGIOR NU
  MERO DI PEZZI
50 REM 2. AGGIUNGE UN VALORE AL
  LE POSIZIONI LATERALI
60 REM LA SCACCHIERA E' IL VETTORE
  A CIRCONDATO DA SPAZI
70 REM A=0 CASELLA VUOTA
80 REM A=8 PER CASELLA NERA -- X
  (INTERAMENTE -1)
90 REM A=W PER CASELLA BIANCA --O
  (INTERAMENTE + 1)
100 REM I E J SEMPRE USATI PER INDI
  CI DI RIGA/COLONNA
110 REM I4 E J4 MEMORIZZANO GLI INC
  REMENTI PER LE OTTO CASELLE VICI
  NE
120 REM C% AE D% MEMORIZZANO I CARA
  TTERI -H,X,.,O PER L'OUTPUT
130 DIM A(100),I4(7),J4(7),X%(100
  )
140 REM APERTURA GIOCO
390 PRINT
400 REM INITIALIZE
410 PRINT "ASPETTO PRIMA DI MUOVERE
  (S/N/O)?";
420 F2=0
430 INPUT X%
440 IF X%(1,1)="N" THEN 480
450 IF X%(1,1)*"S" THEN 430
460 F2=1
480 PRINT "DEVO GIOCARE AL MEGLIO (S
  /N/O)?";
490 S2=0
500 INPUT X%
510 IF X%(1,1)="N" THEN 540
520 IF X%(1,1)*"S" THEN 500
530 S2=2
540 B=-1
550 W=+1
560 I4(1)=0:I4(2)=-1:I4(3)=-1:I4(
  4)=-1:I4(5)=0:I4(6)=1:I4(7)
  =1:I4(8)=1
570 J4(1)=1:J4(2)=1:J4(3)=0:J4(
  4)=-1:J4(5)=-1:J4(6)=-1:J4(
  7)=0:J4(8)=1
580 REM INIZIA UN NUOVO GIOCO
590 FOR A=1 TO 100

```

```

600 A(A)=0
610 NEXT A
620 A(45)=W:A(56)=W:A(46)=B:A(55
  )=B
630 C1=2:H1=2:N1=4:Z*0
640 REM SCELTE DELL'UMANO
650 PRINT "VUOI IL ROSA O IL VERDE?"
  ;:C=W:H=8
660 INPUT X%
670 GR : COLOR=1: VLIN 0,4 AT 1

680 COLOR=2: HLIN 0,2 AT 5: HLIN
  0,2 AT 7: HLIN 0,2 AT 9: PLOT
  2,6: PLOT 0,8
690 COLOR=3: VLIN 10,14 AT 2: HLIN
  0,1 AT 10: HLIN 0,1 AT 14: PLOT
  1,12
700 COLOR=4: VLIN 15,19 AT 2: VLIN
  15,17 AT 0: PLOT 1,17
710 COLOR=5: HLIN 0,2 AT 20: HLIN
  0,2 AT 22: HLIN 0,2 AT 24: PLOT
  0,21: PLOT 2,23
720 COLOR=6: VLIN 25,29 AT 0: VLIN
  27,29 AT 2: HLIN 1,2 AT 25:
  PLOT 1,27: PLOT 1,29
730 COLOR=7: HLIN 0,2 AT 30: VLIN
  31,34 AT 2
740 COLOR=8: VLIN 35,39 AT 0: VLIN
  35,39 AT 2: PLOT 1,35: PLOT
  1,37: PLOT 1,39
750 IF X%(1,1)="R" THEN 760: IF
  X%(1,1)*"O" THEN 660:C=B:H=
  W
760 INPUT "VUOI GIOCARE PER PRIMO (S
  /N/O)?":X%
770 CALL -936
780 GOSUB 2850
790 IF X%(1,1)="N" THEN 850
800 IF X%(1,1)*"S" THEN 780
810 GOTO 1520
820 REM MOSSA AL COMPUTER
830 IF F2=0 THEN 850
840 IF PEEK (-16384)(127 THEN 840
  : POKE -16384,0
850 B1=-25
860 I3=0:J3=0
870 T1=C
880 T2=H
890 REM CERCA UNA CASELLA VUOTA
900 FOR I=1 TO 8
910 FOR J=1 TO 8

```

```

920 GOSUB 2930: IF A#0 THEN I230
930 REM TROVATA UNA CASELLA VUOTA
940 REM HA UN AVVERSAARIO VICINO?
950 GOSUB 2370
960 IF F1=0 THEN I230
970 REM TROVATO L'AVVERSAARIO VICINO

980 REM QUANTI DEI SUOI PEZZI POSSO
    GIRARE?
990 REM (NON FARLO ADESSO)
1000 U=-1
1010 GOSUB 2570
1020 REM PUNTEGGIO EXTRA PER CASELLA
    LATERALE
1030 IF S1=0 THEN I230
1040 IF (I-1)*(I-8)#0 THEN I060
1050 S1=S1+S2
1060 IF (J-1)*(J-8)#0 THEN I080
1070 S1=S1+S2
1080 IF (I-2)*(I-7)#0 THEN I100
1090 S1=S1-S2
1100 IF (J-2)*(J-7)#0 THEN I130
1110 S1=S1-S2
1120 REM E' LA MIGLIORE MOSSA TROVAT
    A?
1130 IF S1<B1 THEN I230
1140 IF S1>B1 THEN I190
1150 REM PARI DECISIONE CASUALE
1160 R=RND (2)
1170 IF R>0 THEN I230
1180 REM YES
1190 B1=S1
1200 I3=I
1210 J3=J

1220 REM FINE DEL LOOP DI RICERCA
1230 NEXT J
1240 NEXT I
1250 REM POSSIAMO FARE QUALCOSA?
1260 IF B1#-25 THEN I340
1270 REM NO
1280 PRINT "DEVO PASSARE..."
1290 FOR X=1 TO 1000: NEXT X
1300 IF Z=1 THEN I980
1310 Z=1
1320 GOTO I520
1330 REM FAI LA MOSSA
1340 Z=0
1350 PRINT "MUCCHIO IN ";I3;" , ";J3
1360 FOR X=1 TO 1000: NEXT X
1370 I=I3
1380 J=J3
1390 U=1
1400 GOSUB 2570
1410 C1=C1+S1+1
1420 H1=H1-S1
1430 NI=NI+1
1440 PRINT "GUADAGNO ";S1;" DEI TUOI
    PEZZI"
1450 REM STAMPA LA SCACCHIERA
1460 GOSUB 2850
1470 FOR X=1 TO 500: NEXT X
1480 REM TEST PER FINE GIOCO
1490 IF H1=0 THEN I980
1500 IF NI=64 THEN I980
1510 REM MOSSA ALL'UMANO
1520 T1=H
1530 T2=C

```

```

1540 PRINT
1550 PRINT "TOCCA A TE.... (RIGA,CO
    L)";
1560 INPUT I,J
1570 IF I<0 THEN I560
1580 IF I>8 THEN I560
1590 IF I#0 THEN I630
1600 PRINT "STAI PASSANDO? (S/NO)?"
    I
1610 INPUT X#
1620 IF X#<1,1>#0 THEN I550
1630 IF Z=1 THEN I980
1640 Z=1
1650 GOTO B30
1660 IF J<1 THEN I560
1670 IF J>8 THEN I560
1680 REM CONTROLLA SE VUOTA
1690 GOSUB 2930: IF A=0 THEN I730

1700 PRINT "CASELLA OCCUPATA"
1710 GOTO I560
1720 REM CONTROLLA SE IL VICINO E' L
    EGALE
1730 GOSUB 2370
1740 IF F1=1 THEN I780
1750 PRINT "MOSSA ERRATA"
1760 GOTO I560
1770 REM
1780 U=-1
1790 GOSUB 2570
1800 IF S1>0 THEN I840
1810 PRINT "MOSSA ERRATA"
1820 GOTO I560
1830 REM TUTTO OK, FAI LA MOSSA
1840 Z=0
1850 PRINT "GUADAGNI ";S1;" DEI MIEI
    PEZZI"
1860 U=1
1870 GOSUB 2570

1880 H1=H1+S1+1
1890 C1=C1-S1
1900 NI=NI+1
1910 REM STAMPA LA SCACCHIERA
1920 GOSUB 2850
1930 REM TEST SE FINE GIOCO
1940 IF C1=0 THEN I980
1950 IF NI=64 THEN I980
1960 GOTO B30
1970 REM FINE GIOCO
1980 PRINT
1990 PRINT "FINE GIOCO-"; TAB 3
2000 IF H1=C1 THEN I2040
2010 IF H1<C1 THEN I2060
2020 PRINT "HO VINTO!!"
2030 GOTO I2070
2040 PRINT "PARI!!"
2050 GOTO I2250
2060 PRINT "GAI VINTO!!"
2070 FOR X=1 TO 1000: NEXT X
2080 C1=C1-H1
2090 C1=ABS (C1)
2100 C1=(64-C1)/NI
2110 PRINT : PRINT "E' STATO UN";

2120 IF C1<11 THEN I2240
2130 IF C1<25 THEN I2220
2140 IF C1<39 THEN I2180
2150 IF C1<53 THEN I2180

```

```

2160 PRINT "GIOCO PERFETTO";
2170 GOTO 2250
2180 PRINT "A PASSEGGIATA";
2190 GOTO 2250
2200 PRINT "VULO";
2210 GOTO 2250
2220 PRINT "GIOCO DURO!!"
2230 GOTO 2250
2240 PRINT "SQUASSO!!";
2250 FOR X=1 TO 1000: NEXT X: PRINT

2260 PRINT "GIOCHI ANCORA (SI/NO)?"
;
2270 INPUT X$
2280 TEXT : CALL -936
2290 IF X$(1,1)="S" THEN 590
2300 IF X$(1,1)="N" THEN 2270
2310 PRINT "GRAZIE PER LA PARTITA"

2320 END
2330 REM SUBROUTINE PER TESTARE IL O
IUSTO VICINO
2340 REM
2350 REM
2360 REM
2370 FOR I=-1 TO 1
2380 FOR J=-1 TO 1
2390 IF A((I+1)*10+J+1)=T2 THEN
2400 NEXT J
2410 NEXT I
2420 REM NON TROVATO
2430 FI=0
2440 RETURN
2450 REM SUCCESSO!!
2460 FI=1
2470 RETURN
2480 REM SUBROUTINE PUNTEGGIO
2490 REM
2500 REM
2510 REM
2520 REM
2530 REM
2540 REM IL MAIN CONTIENE IL SEGUENT
E VETTORE
2550 REM 14: 0 -1 -1 -1 0 1
1 1
2560 REM J4: 1 1 0 -1 -1 -1
0 1
2570 S1=0
2580 FOR K=1 TO B
2590 15=14(K)
2600 J5=J4(K)
2610 16=1+15
2620 J6=J+J5
2630 S3=0
2640 IF A(16*10+J6+1)AT2 THEN 2820

2650 REM LOOP THROUGH THE RUN
2660 S3=S3+1
2670 16=1+15
2680 J6=J+J5
2690 IF A(16*10+J6+1)=T1 THEN 2720

2700 IF A(16*10+J6+1)=0 THEN 2820

2710 GOTO 2660
2720 S1=S1+S3
2730 IF UN1 THEN 2820
2740 REM UPDATE BOARD
2750 16=1
2760 J6=J
2770 FOR K1=0 TO S3
2780 A(16*10+J6+1)=T1
2790 16=16+15
2800 J6=J6+J5
2810 NEXT K1
2820 NEXT K
2830 RETURN
2840 REM SUBROUTINE CHE STAMPA LA SC
ACCHIERA
2850 FOR I=1 TO 8: FOR J=1 TO 8
2860 COLOR=0: IF A(1*10+J+1)=-1 THEN
COLOR=11: IF A(1*10+J+1)=1 THEN
COLOR=12
2870 VLIN 1*5-4,1*5-3 AT 4*J: VLIN
1*5-5,1*5-2 AT 4*J+1: VLIN
1*5-4,1*5-3 AT 4*J+2
2880 NEXT J: NEXT I
2890 POKE 34,20: POKE 32,0: POKE
33,39: VTAB 21: TAB 6: PRINT
"1 2 3 4 5 6 7 8"
: POKE 34,21
2900 POKE 32,0: POKE 33,10: VTAB
22: TAB 1: PRINT "COMPUTER"
: TAB 4: PRINT C1: " ";

2910 POKE 32,29: VTAB 22: TAB 3:
PRINT "UNANO": TAB 5: PRINT
H1: " "
2920 POKE 32,11: POKE 33,17: VTAB
24: TAB 1: RETURN
2930 A=A(1*10+J+1): RETURN
2940 A(1*10+J+1)=A: RETURN

```

Dischi personalizzati



Questa piccola utility vi permetterà di personalizzare i vostri dischetti sostituendo l'infor-

mazione "DISK VOLUME" con una scritta a vostro piacere. La modifica avviene nella parte

di DOS 3.3 residente in memoria di 11 caratteri. Essa potrà apparire in modo normale, oppure in inverso, oppure ancora in caratteri lampeggianti.

Se vorrete rendere definitiva la vostra modifi-

ca dovete inizializzare un nuovo dischetto con un programma HELLO di vostra scelta. Tutte le volte che con questo dischetto verrà eseguito il comando 'CATALOG' apparirà la nuova scritta!!

```

10 HOME
20 REM *****
   *
   * PERSONNALIZZA *
   *
   * I TUOI DISCHETTI *
   *
   *****

60 REM
65 REM
70 REM
72 REM
75 HTAB 10: GOTO 90
80 HTAB 20 - LEN (Z$) / 2: PRINT
   Z$: RETURN
90 A$ = "*****": B$ =
   "*"
   *": Z$ = A
   $: GOSUB 80: Z$ = B$: GOSUB 8
   0: Z$ = "*PERSONALIZZAZIONE*"
   : GOSUB 80: Z$ = B$: GOSUB 80
   : Z$ = "* DI UN DISCHETTO *"
100 GOSUB 80: Z$ = B$: GOSUB 80: Z
   $ = A$: GOSUB 80
150 PRINT
160 PRINT " CON QUESTO PROGRAMMA
   POTRETE PERSONA-"
170 PRINT "LIZZARE I VOSTRI DISC
   HETTI SOSTITUENDO"
180 PRINT "L'INFORMATIONE DISK V
   OLUME CON QUELLO "
190 PRINT "CHE VORRETE FUNZIONA
   MODIFICANDO IN ME-"
200 PRINT "MORIA IL DOS. POTRETE
   POI RENDERE DEFINITIVA LA
   VOSTRA MODIFICA INIZIALIZZA
   N- DO UN DISCHETTO CON UN PR
   OGRAMMA 'HELLO' DI VOSTRA SCE
   LTA."
210 PRINT "LA NUOVA SCRITTA POTR
   A' AVERE UNA LUN- "
220 PRINT "GHEZZA MASSIMA DI 11
   CARATTERI."
240 PRINT " LA SCRITTA PUO' ESSE
   RE:"
250 HTAB 10: PRINT " 1-IN VIDEO
   NORMALE"
260 HTAB 10: PRINT " 2-IN VIDEO
   INVERSO"
270 HTAB 10: PRINT " 3-IN MODO L
   AMPEGGIANTE "
280 PRINT
290 PRINT "SCEGLI:1,2,3": HTAB
   30: INPUT " ? ": A$: A = A$
295 PRINT : PRINT : PRINT
300 IF A < 1 OR A > 3 THEN 290

```

```

310 PRINT : PRINT
320 PRINT "OAI LA TUA MODIFICA"
330 PRINT "(MASSIMO 11 CARATTERI
   )": HTAB 27: INPUT A$
350 X = LEN (A$)
360 IF X > 11 THEN 320
370 B = 11 - X
380 IF A = 1 OR A = 3 THEN B$ =
390 IF A = 2 THEN B$ = ""
395 IF B = 0 THEN 430
400 FOR I = 1 TO B
410 A$ = B$ + A$
420 NEXT I
430 FOR J = 1 TO 11
440 B$ = MID$ (A$, J, 1)
450 IF A = 1 THEN POKE 46011 -
   J, ASC (B$) + 128
460 IF A = 2 THEN POKE 46011 -
   J, ASC (B$) - 64 * (ASC (B$
   ) > 64)
470 IF A = 3 THEN POKE 46011 -
   J, ASC (B$) + 64 * (ASC (B$
   ) < 64)
480 NEXT J
485 HOME
490 D$ = "": REM CTRL-D
500 PRINT D$:"CATALOG"
505 PRINT : PRINT
510 PRINT "VI PIACE LA MODIFICA
   ?"
520 HTAB 10: PRINT "1=SI"
530 HTAB 10: PRINT "2=NO": HTAB
   30: INPUT Z: PRINT : PRINT
540 HOME
545 IF Z < 1 OR Z > 2 THEN 510
550 IF Z = 2 THEN 240
560 PRINT "PER SALVARE QUESTA MO
   DIFICA CARICATE UN PROGRAMMA
   'HELLO' DI VOSTRA SCELTA "
570 PRINT "EO INIZIALIZZATE UN N
   UOVO DISCHETTO"
590 PRINT : PRINT : PRINT "
   CIAO!!!!!"

```

Per ricevere gli arretrati di **PAPER J.soft** compilare il seguente tagliando:

J.soft Compilare ed inviare in busta chiusa a:
viale Restelli, 5 - 20124 Milano - Tel. 02/6888228-683797-6880841-2-3

Nome _____

Cognome _____

Indirizzo _____

CAP _____ Città _____ Prov. _____

Desidero ricevere i seguenti numeri arretrati di

PAPER J.soft _____ Anno _____

PAPER J.soft dedicato a _____ Anno _____

a L. 2.000 cad. per un totale di L. a mezzo

☐ assegno allegato

☐ contanti allegati



TI-99/4A

Extended BASIC

Disassembler

La parola Disassembler ricorda indubbiamente quella giungla di cifre che costituisce il linguaggio Assembler; ma, come ricorda il prefisso, ha il vantaggio di sezionarlo rendendolo più comprensibile.

Il programma in questione non si distacca dai più comuni Disassembler, che permettano l'associazione ad ogni OPCODE (istruzione in LM) di una rappresentazione, in stringa da una o quattro lettere, chiamata MNEMONIC.

Questo Disassembler infatti decodifica il contenuto della memoria in MNEMONICS-TI standard, con l'evidente pregio di rendere il LM più comprensibile.

Insomma questa è una "chicca" per i più affezionato 99isti di Papersoft (quelli che non scambiano il Linguaggio Macchina con la lingua dei meccanici) che offrirà, con l'ausilio della intravabile MINI-MEMORY o del più diffuso (grazie alla importazione J.soft) modulo Extended BASIC con l'ausilio di una espansione RAM di qualsiasi tipo la possibilità di approfondire le conoscenze del proprio computer; va benissimo anche una di quelle espansioni commercializzate di queste tempi, che non abbisognano di Peripheral Expansion Box o simili. Detto questo, buon lavoro!

```

30 REM INIZIALIZZA.
40 TR=0 :: CALL CLEAR :: PR=0
50 PRINT "INDIRIZZO DI PARTENZA
(PMO'ESSERE ANCHE UN DECIMALE)?"
:: INPUT A
60 IF A=0 THEN 80
70 IF A/Z<>INT(A/2) THEN 50
80 PRINT "ENO ADDRESS ?" :: INPU
T B
90 PRINT "DECODIFICA OA : ";A;"
A : ";B
100 PRINT "VUOI STAMPARE?" :: IN
PUT J$
110 IF J$="S" THEN PR=1 :: CLOSE
#1 :: OPEN #1:"RS232",OUTPUT
120 IF A>=B THEN 50
130 A1=A :: IF A>32767 THEN A1*A
-65536
140 CALL PEEK(A1,H1,L1,O1,O2,O3,O4
)
150 REM TEST PER OPCODES & ADDRE
SS MODES
160 N=H*256+L :: IF N>16383 THEN
690
170 IF N>14335 THEN 600
180 IF N>12287 THEN 540
190 IF N>11263 THEN 600
200 IF N>8191 THEN 600
210 IF N>4095 THEN 460
220 IF N>2047 THEN 430
230 IF N>1023 THEN 380
240 IF N>831 THEN 790
250 IF N>511 THEN 270
260 GOTO 820
270 REM FORMATO VIII OP-COOES
280 IF (L AND 16)=16 THEN 820

290 RESTORE 1020 :: J=480 :: Z=5
:: K=32 :: R=(L AND 15):: N=((H
AND 3)*256)+(L AND 224):: GOSUB
900
300 IF TR<>1 THEN 330
310 C$=STR$(O1*256+O2):: A=A+4
320 B$="R"&STR$(R)&" " :: GOTO 3
70
330 Z=2 :: GOSUB 900 :: IF TR<>1
THEN 350
340 C$="" :: A=A+2 :: B$="R"&STR
$(R) :: GOTO 370
350 Z=2 :: GOSUB 900 :: IF TR<>1
THEN 820
360 B$=STR$(O1*256+O3):: A=A+4 :
: C$=""
370 GOSUB 860 :: TR=0 :: GOTO 12
0
380 REM FORMATO VI OP-COOES
390 N=((H*256)+(L AND 192):: J=96
0 :: Z=14 :: K=64 :: RESTORE 100
0 :: GOSUB 900
400 GOSUB 930
410 C$="" :: IF A$="B" AND B$="*
R11" THEN C$="(SAME AS RTS)"

```

```

420 GOSUB 860 :: TR=0 :: GOTO 12
0
430 REM FORMATO V OP-COOES
440 N=((H AND 11)*256 :: J=1792 :
: CO=(L AND 240)/16 :: WR=(L AND
15):: RESTORE 1040 :: Z=4 :: K=
256 :: GOSUB 900
450 B$="R"&STR$(WR)&" " :: C$=ST
R$(CO):: A=A+2 :: GOSUB 860 :: T
R=0 :: GOTO 120
460 REM FORMAT II OP-COOES
470 RESTORE 1050 :: J=3840 :: TR
=0 :: Z=13 :: K=256 :: N=H*256 :
: GOSUB 900
480 IF TR=0 THEN 500
490 IF L>127 THEN L=L-256

490 B$=STR$(2*L+2):: GOTO 520
500 Z=3 :: K=256 :: GOSUB 900
510 B$=STR$(L)
520 C$="" :: A=A+2 :: IF A$="JMP
" AND B$="2" THEN C$="(SAME AS N
OP)"
530 GOSUB 860 :: TR=0 :: GOTO 12
0
540 REM FORMATO IV OP-COOES
550 IF ((H AND 252)=48 THEN A$="L
DCR" :: GOTO 580
560 IF ((H AND 252)=52 THEN A$="S
TCK" :: GOTO 580
570 GOTO 820
580 GOSUB 930 :: NU=((H AND 3)*4
)+(L AND 192)/64)
590 C$=","&STR$(NU):: GOSUB 860
:: TR=0 :: GOTO 120
600 REM FORMATO III & IX OP-CODE
S
610 RESTORE 1070 :: J=7168 :: N=
((H AND 60)*256):: D=((H AND 3)*
4)+(L AND 192)/64:: Z=3 :: K=1
024 :: GOSUB 900
620 IF TR<>1 THEN 650
630 C$=","&"R"&STR$(O1
640 GOSUB 930 :: GOSUB 860 :: TR
=0 :: GOTO 120
650 IF N<>11264 THEN 670
660 A$="XOP" :: C$="XOP OP #"&ST
R$(D) :: GOTO 640
670 J=13312 :: Z=2 :: K=1024 ::
GOSUB 900 :: IF TR<>1 THEN 820
680 GOTO 630
690 REM FORMATO I OP-COOES
700 RESTORE 1080 :: J=12288 :: N
=((H AND 240)*256 :: Z=12 :: K=40
961:: GOSUB 900
710 TO=((H AND 12):: D=((H AND 3)
*4)+(L AND 192)/64):: GOSUB 930

720 IF TD=0 THEN C$=","&"R"&STR$
(D) :: GOTO 770
730 IF TD=4 THEN C$=","&"*R"&STR
$(D) :: GOTO 770

```

```

740 IF TD=12 THEN C$=",*R"&STR$(
D)&"+" :: GOTO 770
750 IF (TD=8) AND (D=0) THEN C$=","
&"@&STR$(01*256+02):: A=A+2 ::
IF TS=32 THEN C$=","&"@&STR$(03
*256+04)
760 IF (TO=8) AND (O<>0) THEN C$=","
&"@&STR$(01*256+02)&"(R"&STR$(S)&
")" :: A=A+2 :: IF TS=32 THEN C$
=","&"@&STR$(03*256+04)&"(R"&STR$(
S)&"")
770 GOSUB 860 :: TR=0 :: GOTO 12
0
780 IF (TO=8) AND (O=0) THEN C$=","&
"&STR$(01*256+02):: A=A+2 :: IF
TS=32 THEN C$=","&"@&STR$(03*256+0
4)
790 REM FORMATO VII
800 N=(H*256+L):: A=A+2 :: B$=""
:: C$="" :: Z=6 :: J=800 :: K=3
2 :: RESTORE 1100
810 GOSUB 900 :: GOSUB 860 :: TR
=0 :: GOTO 120
820 REM NESSUN OP-CODE
830 A$="BYTE" :: B$=STR$((H*256)
+L):: C$=CHR$(H)&" "&CHR$(L):: A
=A+2 :: GOSUB 860
840 REM CALL SOUND(800,400,0)::
ACCEPT Q4$ :: GOTO 97
850 GOTO 120
860 REM ROUTINE STAMPA
870 PT$=STR$(A1)&" "&"A$&" "&B$&
C$ :: PRINT PT$ :: A1=A
880 IF PR=1 THEN PRINT #1;TAB(10
);PT$
890 RETURN
900 REM RICERCA OP-CODE DAI DATA

```

```

910 FOR I=1 TO Z :: J=J+K :: REA
D Q$ :: IF N=J THEN A$=Q$ :: TR=
1
920 NEXT I :: RETURN
930 REM SUBROUTINE TS ADDRESS
940 S=(L AND 15):: TS=(L AND 48)
:: IF TS=0 THEN B$="R"&STR$(S)::
A
A+2 :: RETURN
950 IF TS=16 THEN B$="R"&STR$(S
):: A=A+2 :: RETURN
960 IF TS=48 THEN B$="R"&STR$(S
)&"+" :: A=A+2 :: RETURN
970 IF (TS=32) AND (S=0) THEN B$="@
"&STR$(01*256+02):: A=A+4 :: RET
URN
980 IF (TS=32) AND (S<>0) THEN B$="
@&"&STR$(01*256+02)&"(R"&STR$(S)&
")" :: A=A+4 :: RETURN
990 BREAK
1000 OATA BLWP,B,X,CLR,NEG,INV,I
NC,INCT,DEC
1010 OATA DECT,BI,SWPB,SETO,ABS

1020 OATA LI,A1,ANDI,ORI,C1
1030 OATA STWP,STST,LWPI,LIMI
1040 OATA SRA,SRL,SLA,SLR
1050 DATA JMP,JLT,JLE,JEQ,JHE,JG
T,JNE
1060 DATA JNC,JOC,JNO,JL,JH,JOP,
SBO,SBZ,TB
1070 OATA COC,CZC,XOR,MPY,DIV
1080 DATA SZC,SZCB,S,SB,C,CB,A,A
B
1090 OATA MOV,MOVE,SOC,SOCB
1100 DATA IOLE,RSET,RTWP,CKON,CK
OF,LREX

```

Tank



TI-99/4A

Anche se il suono può ricordare quello di un'altra parola inglese che ha a che vedere con il ringraziamento, il titolo di questo gioco se ne distacca del tutto: si tratta di guerra; in particolare di carri-armati!

Vostro compito sarà quello di difendere un ponte dell'avanzata dei carri nemici; per cui do-

vrete sparare su di essi senza pietà (al minimo sbaglio quelli si voltano e BUM! Siete fregati!) Per fare ciò, premete la barra spaziatrice, insieme al tasto (E) per tiri a lunga gittata; ed insieme al tasto (X) per cose più modeste. Detto questo, forza! E in bocca al Carro!

```

10 REM *****
20 REM ** TANK **
23 REM *****
26 REM TI-99 BASIC
30 FOR COL=2 TO 12
40 CALL COLOR(COL,COL+1,COL+1)
50 NEXT COL

60 CALL SCREEN(2)
70 CALL CLEAR
80 PRINT "{(00 888 h H
AA A"
90 PRINT "PPXX h h pp X
PP P"
100 PRINT " xx 1 1 1 0
{ { "
110 PRINT " 00 8 8 8 A p
AA "
120 PRINT " HH P PH H p
X{ ( "
130 PRINT " `` hYFMh A A 0
88 8 "
140 PRINT " xx 1 1 0 00
{ { "
150 PRINT " 00 8 8 H H
AA A"
160 PRINT ::::::::::
170 FOR I=1 TO 500
180 NEXT I
190 CALL CLEAR
200 CALL SCREEN(16)
210 FOR I=1 TO 12
220 CALL COLOR(I,2,1)
230 NEXT I
240 PRINT "VUOI LE ISTRUZIONI?(S
/N)"
250 CALL KEY(0,K,S)
260 IF K<1 THEN 250
270 IF K=78 THEN 290 ELSE 2420
280 GOTO 250
290 LIFE=3
300 CALL CHAR(54,"FFFF7E7E3C3C18
18")
310 CALL CHAR(55,"3C3C3C3C3C3C3C
18")
320 DIM P(21)
330 CALL COLOR(9,9,1)
340 DIM R(24)
350 SHOT=1
360 CALL COLOR(8,5,1)
370 CALL COLOR(1,4,4)
380 CALL SCREEN(8)
390 SCORE=0
400 CALL COLOR(13,7,15)
410 CALL COLOR(15,3,15)
420 CALL COLOR(16,3,15)
430 CALL COLOR(12,15,15)
440 CALL CHAR(120,"FFFFFFFFFFFFF
FFFFFFF")
450 CALL COLOR(11,13,15)
460 CALL COLOR(2,2,15)
470 Y=5
480 G=1

```

```

490 RANDOMIZE
500 CALL CHAR(113,"000000FF00000
00F")
510 CALL CHAR(112,"3C7EFFFFFFF7
FFF")
520 CALL CHAR(145,"0000E0F0F0E08
0FF")
530 CALL CHAR(146,"FF7F3F1F0F070
301")
540 CALL CHAR(152,"FFFFFFFFFAA55A
AFF")
550 CALL CHAR(153,"FEFFFFFFFEFCF
8F0")
560 CALL CHAR(154,"000080")
570 CALL CHAR(101,"18183C3C7E7E
FFF")
580 CALL CHAR(91,"183C3C3C3C3C3C
3C")
590 CALL CLEAR
600 CALL CHAR(135,"FFFFFFFFFFFFFF
FFFF")
610 REM PONTE
620 CALL HCHAR(11,1,135,32)
630 CALL VCHAR(12,8,135,4)
640 CALL VCHAR(12,16,135,4)
650 CALL VCHAR(12,24,135,4)
660 CALL VCHAR(12,32,135,4)
670 CALL HCHAR(8,1,45,32)
680 CALL HCHAR(7,1,120,32)
690 CALL HCHAR(6,1,120,32)
700 CALL HCHAR(4,1,120,32)
710 CALL HCHAR(3,1,120,32)
720 CALL HCHAR(9,1,120,32)
730 CALL HCHAR(10,1,120,32)
740 CALL HCHAR(5,1,45,32)
750 CALL HCHAR(2,1,45,32)
760 CALL HCHAR(1,1,135,32)
770 CALL CHAR(137,"7E7E7E7E7E7E7
E7E")
780 CALL CHAR(136,"3C3C3C3C3C3C3C
3C")
790 CALL CHAR(138,"FFFFFFFFFFFFF
FFF")
800 CALL COLOR(14,13,4)
810 GOTO 1700
820 CALL KEY(0,M,S)
830 IF M=69 THEN 890
840 IF M=88 THEN 930
850 IF M=83 THEN 970
860 IF M=68 THEN 1010
870 IF M=32 THEN 1060
880 GOTO 1810
890 G=G+1
900 IF G>3 THEN 910 ELSE 920
910 G=3
920 GOTO 1360
930 G=G-1
940 IF G<1 THEN 950 ELSE 960
950 G=1
960 GOTO 1360
970 Y=Y-1
980 IF Y<3 THEN 990 ELSE 1370

```

```

990 Y=3
1000 GOTO 1360
1010 Y=Y+1
1020 IF Y>27 THEN 1030 ELSE 1360

1030 Y=27
1040 GOTO 1360
1050 REM SPARI GIOCATORE

1060 IF SHOT=0 THEN 1360
1070 CALL GCHAR(T,Y+2,XX)
1080 FOR I=F TO T STEP -1
1090 CALL GCHAR(I+1,Y+2,R(I))
1100 NEXT I
1110 CALL VCHAR(T,Y+2,101,LE)
1120 CALL VCHAR(T,Y+2,91)
1130 FOR J=F TO T STEP -1
1140 CALL VCHAR(J+1,Y+2,R(J))
1150 NEXT J
1160 CALL SOUND(1,1760,0)
1170 SHOT=SHOT-1
1180 IF XX=112 THEN 1200 ELSE 13
30
1190 CALL HCHAR(I+1,Y+2,120)
1200 SCORE=SCORE+20
1210 AS="SCORE"&STR$(SCORE)
1220 CALL COLOR(3,2,1)
1230 FOR I=1 TO LEN(AS)
1240 CALL HCHAR(1,I+2,ASC(SEG$(A
$,I,1)))
1250 NEXT I
1260 FOR I=1 TO 16
1270 CALL SCREEN(I)
1280 CALL SOUND(I,110*I,I)
1290 NEXT I
1300 CALL SCREEN(8)
1310 GOTO 1950
1320 XX=0
1330 CALL SOUND(200,-5,0)
1340 CALL HCHAR(I+1,Y+2,120)
1350 I=0
1360 CALL VCHAR(16,Y+2,32,8)
1370 CALL VCHAR(21,Y,138,4)
1380 CALL VCHAR(21,Y-1,32,4)
1390 CALL VCHAR(21,Y+4,138,4)
1400 CALL VCHAR(21,Y+5,32,4)
1410 CALL VCHAR(16,Y+1,32,9)
1420 CALL VCHAR(16,Y+3,32,9)
1430 REM MOVIMENTO GIOCATORE

1440 ON G GOTO 1450,1530,1610
1450 CALL HCHAR(24,Y+2,138)
1460 F=20
1470 T=9
1480 LE=11
1490 CALL HCHAR(23,Y+2,137)
1500 CALL HCHAR(22,Y+2,136)
1510 IF Y+1=HOR THEN 2070
1520 GOTO 1810
1530 CALL VCHAR(23,Y+2,138,2)
1540 F=17
1550 T=6
1560 LE=11

```

```

1570 CALL VCHAR(21,Y+2,137,2)
1580 CALL VCHAR(19,Y+2,136,2)
1590 IF Y+1=HOR THEN 2070
1600 GOTO 1810
1610 CALL VCHAR(22,Y+2,138,3)
1620 LE=11
1630 T=3

1640 F=13
1650 CALL VCHAR(19,Y+2,137,3)
1660 CALL VCHAR(16,Y+2,136,3)
1670 IF Y+1=HOR THEN 2070
1680 GOTO 1810
1690 REM TRAIETTORIA NEMICO
1700 K=INT(RND*3)+1
1710 IF K=1 THEN 1740
1720 IF K=2 THEN 1760
1730 IF K=3 THEN 1780
1740 VERT=9
1750 GOTO 1790
1760 VERT=6
1770 GOTO 1790
1780 VERT=3
1790 HOR=29
1800 REM MOVIMENTO NEMICO
1810 HOR=HOR-1

1820 IF HOR=2 THEN 1950 ELSE 184
0
1830 GOTO 1950
1840 CALL HCHAR(VERT,HOR-1,113)

1850 CALL HCHAR(VERT,HOR,112)
1860 CALL HCHAR(VERT,HOR+1,145)

1870 CALL HCHAR(VERT,HOR+2,120)

1880 CALL HCHAR(VERT+1,HOR-1,146
)
1890 CALL HCHAR(VERT+1,HOR,152)

1900 CALL HCHAR(VERT+1,HOR+1,153
)
1910 CALL HCHAR(VERT+1,HOR+2,154
)
1920 CALL HCHAR(VERT+1,HOR+3,120
)
1930 IF Y+1=HOR THEN 2070
1940 GOTO 820
1950 CALL HCHAR(VERT,HOR+1,120)

1960 CALL HCHAR(VERT,HOR+2,120)

1970 CALL HCHAR(VERT,HOR,120)
1980 CALL HCHAR(VERT,HOR-1,120)

1990 CALL HCHAR(VERT+1,HOR+1,120
)
2000 CALL HCHAR(VERT+1,HOR+2,120
)
2010 CALL HCHAR(VERT+1,HOR+3,120
)
2020 CALL HCHAR(VERT+1,HOR,120)

```

```

2030 CALL HCHAR(VERT+1,HOR-1,120
)
2040 SHOT=1
2050 GOTO 1700
2060 REM SPARO NEMICO
2070 CALL HCHAR(VERT,HOR-1,120)

```

```

2080 CALL CHAR(114,"3C7E9FCPCFF
7FFF")

```

```

2090 CALL HCHAR(VERT,Y+1,114)
2100 CALL HCHAR(VERT,Y+2,120)

```

```

2110 LE=0
2120 FOR I=VERT+2 TO 21
2130 LE=LE+1
2140 CALL GCHAR(I,Y+2,P(I))
2150 NEXT I
2160 CALL GCHAR(Y+2,21,B)
2170 CALL COLOR(3,10,1)
2180 CALL VCHAR(VERT+2,Y+2,54,LE
)
2190 CALL HCHAR(21,Y+2,55)
2200 FOR I=VERT+2 TO 21

```

```

2210 CALL HCHAR(I,HOR+1,P(I))
2220 NEXT I
2230 CALL COLOR(3,2,1)
2240 LE=0
2250 LIFE=LIFE-1
2260 FOR I=1 TO 10
2270 CALL SOUND(10*I,110*I,1)
2280 NEXT I
2290 IF LIFE<? THEN 2310
2300 GOTO 1810
2310 CALL CLEAR
2320 FOR I=1 TO 8
2330 CALL COLOR(I,2,1)
2340 NEXT I

```

```

2350 PRINT "ALLA FINE SEI STATO
COLPITO DA UN TANK NEMICO!": "HAI
FATTO ";SCORE;"PUNTI."
2360 PRINT "UN'ALTRA PARTITA? {S
/N}"
2370 CALL KEY(0,K,S)
2380 IF K=83 THEN 290
2390 IF K=78 THEN 2410
2400 GOTO 2370

```

```

2410 END
2420 CALL CLEAR
2430 PRINT
2440 PRINT
2450 PRINT "CONTROLLANDO UN CARR
ARMATO, LA TUA MISSIONE E' DI": "
SPARARE AI CARRI NEMICI ": "IMPED
ENDO LORO DI ATTRAVER- SARE IL P
ONTE."
2470 PRINT "PREMI LA BARRA PER S
PARARE."
2480 PRINT
2490 PRINT "<S> TI MUOVE A SINIS
TRA": "E <D> A DESTRA.": "<E> ALLU
NGA LA GITTATA DEI ": "TUOI TIRI.
"

```

```

2500 PRINT "<X> COMPIE L'OPPOSTO
."
2510 PRINT
2520 PRINT "PUOI SPARARE UN COLP
O ALLA VOLTA E DEVI MIRARE ALLA
TORRETTA.": "SE SBAGLI VERRAI
MIRATO A TUA VOLTA."
2530 PRINT "FORTUNATAMENTE HAI 3
VITE..."
2540 PRINT
2550 PRINT "PREMI UN TASTO..."
2560 CALL KEY(0,K,S)
2570 IF K>0 THEN 290 ELSE 2560

```

Solitario



TI-99/4A

Quella di questa settimana è una bellissima versione in TI-BASIC di uno dei più antichi giochi di pazienza e strategia, conosciuto sotto diversi nomi; una versione è riportata addirittura sul package della Texas intitolato "Oldies but Goodies".

Scopo del gioco è spostare a una a una, verticalmente o orizzontalmente, le 32 pedine (disposte sulla scocchiera a croce di 33 caselle)

in modo da "mangiare" quella che viene saltata con un "passo" di due caselle.

Logicamente, in questo gioco con qualche affinità con la celebre dama, lo scopo è restare con il minor numero possibile di pedine "immangiabili"; se reسترete senza possibili mosse legali e tenderete di farne qualcuna illegale sarà il TI stesso a redorguiri, quindi a buon barare!

```

5 REM *** TI-99 BASIC ***
10 CALL CLEAR
20 PRINT " S O L I T A R I O "
: " P A P E R S O F T " : : : :
: : : :
30 P$="0000001111000000"
40 CALL CHAR(121,P$)
50 CALL COLOR(12,16,16)
60 CALL HCHAR(11,1,121,32)
70 CALL HCHAR(19,1,121,32)
90 PRINT : "VUOI LE ISTRUZIONI
2": "DIGITA, PER FAVORE, -S-O-N-,
"
100 CALL KEY(0,K,S)
110 IF S=0 THEN 100

120 IF K=83 THEN 140
130 IF K=78 THEN 230
140 CALL CLEAR
150 PRINT "AVRAI A DISPOSIZIONE,
": "INIZIALMENTE, 32 PALLINE.":

160 PRINT "PER SPOSTARLE SULLA T
AVOLA": "DEVI FORNIRE LE COORDINA
TE": "DI ORIGINE E DI DESTINAZION
E":
170 PRINT "LE MOSSE ILLEGALI NON
": "VERRANNO COMPUTE.":
180 PRINT "SCOPO DEL GIOCO E'":
LASCIARE ISOLATO ": "IL MINOR NUM
ERO": "DI PEDINE": "BUONA FORTUNA
!": : : :
190 PRINT "PREMI UN TASTO"
200 CALL KEY(0,K,S)
210 IF S=0 THEN 200
220 IF K>32 THEN 230
230 REM
240 CALL CLEAR
250 CALL COLOR(5,2,16)
260 CALL COLOR(12,16,16)
270 CALL COLOR(6,2,16)
280 CALL COLOR(3,2,16)
290 CALL COLOR(4,2,16)
300 CALL COLOR(7,2,16)
310 CALL COLOR(8,2,16)
320 CALL HCHAR(1,9,120,1)
330 CALL HCHAR(17,9,120,1)
340 CALL HCHAR(1,10,65,1)
350 CALL HCHAR(17,10,65,1)
360 CALL HCHAR(1,11,120,1)
370 CALL HCHAR(17,11,120,1)
380 CALL HCHAR(1,12,66,1)
390 CALL HCHAR(17,12,66,1)
400 CALL HCHAR(1,13,120,1)
410 CALL HCHAR(17,13,120,1)
420 CALL HCHAR(1,14,67,1)
430 CALL HCHAR(17,14,67,1)
440 CALL HCHAR(1,15,120,1)
450 CALL HCHAR(17,15,120,1)
460 CALL HCHAR(1,16,68,1)
470 CALL HCHAR(17,16,68,1)

480 CALL HCHAR(1,17,120,1)
490 CALL HCHAR(17,17,120,1)

500 CALL HCHAR(1,18,69,1)
510 CALL HCHAR(17,18,69,1)
520 CALL HCHAR(1,19,120,1)
530 CALL HCHAR(17,19,120,1)
540 CALL HCHAR(1,20,70,1)
550 CALL HCHAR(17,20,70,1)
560 CALL HCHAR(1,21,120,1)
570 CALL HCHAR(17,21,120,1)
580 CALL HCHAR(1,22,71,1)
590 CALL HCHAR(17,22,71,1)
600 CALL HCHAR(1,23,120,1)
610 CALL HCHAR(17,23,120,1)
620 CALL VCHAR(1,8,120,2)

630 CALL VCHAR(1,24,120,2)
640 CALL VCHAR(3,8,49,1)
650 CALL VCHAR(3,24,49,1)
660 CALL VCHAR(4,8,120,1)
670 CALL VCHAR(4,24,120,1)
680 CALL VCHAR(5,8,50,1)
690 CALL VCHAR(5,24,50,1)
700 CALL VCHAR(6,8,120,1)
710 CALL VCHAR(6,24,120,1)
720 CALL VCHAR(7,8,51,1)
730 CALL VCHAR(7,24,51,1)
740 CALL VCHAR(8,8,120,1)
750 CALL VCHAR(8,24,120,1)
760 CALL VCHAR(9,8,52,1)
770 CALL VCHAR(9,24,52,1)
780 CALL VCHAR(10,8,120,1)
790 CALL VCHAR(10,24,120,1)
800 CALL VCHAR(11,8,53,1)
810 CALL VCHAR(11,24,53,1)
820 CALL VCHAR(12,8,120,1)
830 CALL VCHAR(12,24,120,1)
840 CALL VCHAR(13,8,54,1)
850 CALL VCHAR(13,24,54,1)
860 CALL VCHAR(14,8,120,1)
870 CALL VCHAR(14,24,120,1)
880 CALL VCHAR(15,8,55,1)
890 CALL VCHAR(15,24,55,1)
900 CALL VCHAR(16,8,120,1)
910 CALL VCHAR(16,24,120,1)
920 CALL VCHAR(17,8,120,1)
930 CALL VCHAR(17,24,120,1)
940 REM
950 CALL CHAR(96,"3C7EFFFFFFFF7E
3C")
960 CALL COLOR(9,2,1)
970 CALL HCHAR(22,1,32,32)
980 FOR Z=3 TO 15 STEP 2
990 FOR X=14 TO 18 STEP 2
1000 CALL HCHAR(Z,X,96,1)
1010 NEXT X
1020 NEXT Z
1030 FOR Z=7 TO 11 STEP 2
1040 FOR S=10 TO 12 STEP 2
1050 CALL VCHAR(Z,S,96,1)
1060 NEXT S
1070 NEXT Z
1080 FOR Z=7 TO 11 STEP 2
1090 FOR S=20 TO 22 STEP 2
1100 CALL VCHAR(Z,S,96,1)

```

```

1110 NEXT S
1120 NEXT Z
1130 CALL COLOR(2,2,1)
1140 CALL HCHAR(9,16,43,1)
1150 EINS="DA A "
1160 FOR K=1 TO 15
1170 CALL HCHAR(21,K+7,ASC(SEGS(
EINS,K,1)))
1180 NEXT K
1190 P=32.
1200 REM

```

```

1210 CALL HCHAR(21,12,32,2)
1220 CALL HCHAR(21,20,32,2)
1230 CALL KEY(0,KEY,STATUS)
1240 IF STATUS=0 THEN 1230
1250 IF KEY<65 THEN 1230
1260 IF KEY>71 THEN 1230
1270 VS=CHR$(KEY)
1280 CALL HCHAR(21,12,KEY,1)
1290 CALL KEY(0,KEY,STATUS)
1300 IF STATUS=0 THEN 1290
1310 IF KEY<49 THEN 1290
1320 IF KEY>55 THEN 1290
1330 VI$=CHR$(KEY)
1340 CALL HCHAR(21,13,KEY,1)
1350 CALL KEY(0,KEY,STATUS)
1360 IF STATUS=0 THEN 1350
1370 IF KEY<65 THEN 1350
1380 IF KEY>71 THEN 1350
1390 NS=CHR$(KEY)
1400 CALL HCHAR(21,20,KEY,1)
1410 CALL KEY(0,KEY,STATUS)

```

```

1420 IF STATUS=0 THEN 1410
1430 IF KEY<49 THEN 1410
1440 IF KEY>55 THEN 1410
1450 NI$=CHR$(KEY)
1460 CALL HCHAR(21,21,KEY,1)
1470 REM
1480 A=ASC(VS)
1490 B=ASC(VI$)
1500 C=ASC(NS)
1510 D=ASC(NI$)
1520 IF A=65 THEN 1560
1530 IF A=66 THEN 1560
1540 IF A=70 THEN 1560
1550 IF A=71 THEN 1560 ELSE 1600

```

```

1560 IF B=49 THEN 1210
1570 IF B=50 THEN 1210
1580 IF B=54 THEN 1210
1590 IF B=55 THEN 1210 ELSE 1600

```

```

1600 IF C=65 THEN 1640
1610 IF C=66 THEN 1640
1620 IF C=70 THEN 1640
1630 IF C=71 THEN 1640 ELSE 1680

```

```

1640 IF D=49 THEN 1210
1650 IF D=50 THEN 1210
1660 IF D=54 THEN 1210
1670 IF D=55 THEN 1210 ELSE 1680

```

```

1680 IF D=B-2 THEN 1690 ELSE 170
0
1690 IF A=C THEN 1770 ELSE 1210
1700 IF D=B+2 THEN 1710 ELSE 172
0
1710 IF A=C THEN 1770 ELSE 1210
1720 IF C=A-2 THEN 1730 ELSE 174
0
1730 IF B=D THEN 1770 ELSE 1210

```

```

1740 IF C=A+2 THEN 1750 ELSE 121
0
1750 IF B=D THEN 1770 ELSE 1210

```

```

1760 REM
1770 S=8
1780 FOR I=65 TO 71
1790 S=S+2
1800 IF A=I THEN 1820
1810 NEXT I
1820 Z=1
1830 FOR I=49 TO 55
1840 Z=Z+2
1850 IF B=I THEN 1870
1860 NEXT I

```

```

1870 CALL HCHAR(Z,S,43,1)
1880 CALL SOUND(100,294,0)
1890 REM
1900 IF A>C THEN 1910 ELSE 1930

```

```

1910 CALL HCHAR(Z,S-2,43,1)
1920 CALL SOUND(100,294,0)
1930 IF A<C THEN 1940 ELSE 1960

```

```

1940 CALL HCHAR(Z,S+2,43,1)
1950 CALL SOUND(100,294,0)
1960 IF B>D THEN 1970 ELSE 1990

```

```

1970 CALL HCHAR(Z-2,S,43,1)
1980 CALL SOUND(100,294,0)
1990 IF B<D THEN 2000 ELSE 2020

```

```

2000 CALL HCHAR(Z+2,S,43,1)
2010 CALL SOUND(100,294,0)
2020 REM
2030 S=8
2040 FOR I=65 TO 71
2050 S=S+2
2060 IF C=I THEN 2080

```

```

2070 NEXT I
2080 Z=1
2090 FOR I=49 TO 55
2100 Z=Z+2
2110 IF D=I THEN 2130
2120 NEXT I
2130 CALL HCHAR(Z,S,96,1)
2140 CALL SOUND(100,220,0)
2150 P=P-1
2160 IF P=1 THEN 2170 ELSE 1210

```

```

2170 IF C=68 THEN 2180
2180 IF D=52 THEN 2190 ELSE 2400

2190 CALL HCHAR(20,1,32,32)
2200 FOR A=1 TO 10
2210 CALL SOUND(-100,660,0)
2220 NEXT A
2230 FOR A=1 TO 10
2240 CALL SOUND(-100,660,0,550,0)
2250 NEXT A
2260 FOR A=1 TO 10
2270 CALL SOUND(-100,660,0,550,0,440,0)
2280 NEXT A
2290 FOR A=0 TO 30 STEP 2
2300 CALL SOUND(-100,660,A,550,A,440,A)
2310 NEXT A
2320 HERZ$="          MEGLIO DI COSI'."
2330 CALL HCHAR(21,1,32,32)
2340 FOR K=1 TO 23
2350 CALL HCHAR(21,K+3,ASC(SEGS(HERZ$,K,1)))
2360 NEXT K
2370 CALL HCHAR(21,14,120,1)
2380 CALL HCHAR(21,27,120,4)
2390 GOSUB 2510
2400 CALL HCHAR(21,1,32,32)
2410 CALL HCHAR(21,4,121,27)
2420 SIE$="NON C'E' NESSUN VINCITORE."
2430 FOR K=1 TO 25
2440 CALL HCHAR(21,K+3,ASC(SEGS(SIE$,K,1)))

```

```

2450 NEXT K
2460 CALL HCHAR(21,6,120,1)
2470 CALL HCHAR(21,1,120,1)
2480 CALL HCHAR(21,17,120,1)
2490 CALL HCHAR(21,22,120,1)

2500 GOSUB 2510
2510 REM
2520 CALL HCHAR(22,4,120,27)
2530 NEU$="          VUOI FARE UN' ALTRA"
2540 DRU$="          PARTITA? (<S>/<N>)"

2550 FOR L=1 TO 27
2560 CALL HCHAR(23,L+3,ASC(SEGS(NEU$,L,1)))
2570 NEXT L
2580 CALL HCHAR(23,1,120,4)
2590 CALL HCHAR(23,30,120,2)
2620 CALL HCHAR(24,4,121,27)
2630 FOR M=1 TO 20

2640 CALL HCHAR(24,M+3,ASC(SEGS(DRU$,M,1)))
2650 NEXT M
2660 CALL HCHAR(24,1,120,4)
2670 CALL HCHAR(24,30,120,2)
2690 CALL KEY(0,K,S)
2700 IF S=0 THEN 2690
2710 CALL HCHAR(21,1,32,32)
2720 CALL HCHAR(23,1,32,32)
2730 CALL HCHAR(24,1,32,32)
2740 IF K=83 THEN 950
2750 IF K=78 THEN 2760
2760 CALL CLEAR
2770 STOP

```

Paratrooper



L'extended BASIC

Questo programma vi farà passare le eventuali vertigini (o forse ve le aggraverà!) Scritto in Extended BASIC, sfrutta al meglio le capacità grafiche del TI in un gioco emozionante che non richiede sola fortuna o prontezza di riflessi ma soprattutto calcolo e sangue freddo.

Con due opzioni di gioco possibili, dovrete far atterrare la maggior parte possibile delle vostre truppe aerea-trasportate sulle tre piazzole

in mezzo all'oceano.

In questa esercitazione di paracadutismo, per evitare spiacevoli bagni in mare, dovrete però tener conto della velocità del vento e soprattutto (udite-udite!) del peso di ciascun paracadutista! (Ed, in verità, avrete a disposizione dei soldati un po' piccioni!)

Le istruzioni per il gioco sono incluse nel programma.

Buon ammaraggio!

```

6 !*****
7 !** PARATROOPER **
8 !*****
9 !TI-99**EXT.BASIC
20 CALL CLEAR
30 FOR T=10 TO 14 :: FOR I=10 TO
14 :: DISPLAY AT(12,9):"PARATRO
OPHR"
40 CALL SCREEN(T):: NEXT I :: NE
XT T

50 CALL CHAR(131,"183C7EC3183C18
18"): CALL SCREEN(12)
60 FOR T=450 TO 550 STEP 50 :: F
OR I=9 TO 19
70 DISPLAY AT(14,I):CHRS(131)
80 CALL SOUND(10,T,3):: NEXT I :
: NEXT T
90 FOR I=1 TO 100 :: NEXT I :: G
OSUB 920
100 DISPLAY AT(22,2):"VUOI LE IS
TRUZIONI?(S/N)"
110 ACCEPT AT(22,26)VALIDATE("SN
sn"):YS
120 IF (YS="S")OR(YS="s")THEN 75
0
130 IF (YS="N")OR(YS="n")THEN 86
0
140 CALL CLEAR :: CALL SCREEN(8)

150 CALL CHAR(33,"E7A424E7E781A5
E7",34,"E78585E5E525A5E7")
160 CALL CHAR(37,"F7941417276141
47",42,"503D7C7C7C7A0088U")

170 CALL CHAR(43,"183C7DC3000000
00",44,"08183878F808FF7E")
180 CALL CHAR(46,"187E5A183C0000
00",98,"01031FFFFFFFFF")
190 CALL CHAR(99,"80C0FCFDFDFDFEF
FFF",107,"FFFFFFFFFFFFFFFF")

200 CALL CHAR(117,"FFFFFFFFFFFFFF
FFF",122,"00E0A6E6A6FEBABE")
210 CALL CHAR(130,"00000173FFFD
000",133,"FFFFFFFFFFFF0000")
220 CALL CHAR(134,"FCFCFCFCFCFC
000",135,"FFFFFFFFFFFF0000")
230 CALL CHAR(137,"183C7B7EFFFFF
1818",143,"0E5FFE7F3BC0800")
240 CALL SCREEN(8):: CALL COLOR(
9,4,8,10,6,1)
250 CALL HCHAR(16,1,107,2561)
260 FOR I=1 TO 31 STEP 2 :: CALL
HCHAR(16,I,98):: CALL HCHAR(16,
I+1,99):: NEXT I
270 POINT=0 :: PARA=10
280 RANDOMIZE :: FOR N=22 TO 24
:: G=INT(RND*100)+10
290 CALL SPRITE(#N,143,15,G,G+12
0,0,.60):: NEXT N
300 S=7 :: FOR N=4 TO 6 :: S=S-1
:: RANDOMIZE

```

```

310 D=INT(RND*5)+1 :: DD=INT(RND
*14)+3 :: IF (D=OD)+(DD=ODD)+(DD
=6)THEN 310
320 OD=D :: ODD=DD :: J=N*10+90+
RND*10 :: CALL SPRITE(#S,44,DD,J
,J,0,D1):: NEXT N
330 IF FL=1 THEN 370 ELSE DISPLA
Y AT(15,5):CHRS(37):: DISPLAY AT
(15,14):CHRS(34)

340 DISPLAY AT(15,23):CHRS(33)
350 CALL SPRITE(#3,32,1,180,180,
0,60):: !SPRITE INVISIBILE NELL'
OCEANO
360 CALL SPRITE(#7,133,10,121,19
3,#8,135,12,121,121,#9,134,14,12
1,49):: !PIAZZOLE
370 IF PARA=0 THEN 630 ELSE RAND
OMIZE :: U=INT(RND*70)+10 :: !VO
LO AEREO
380 CALL SPRITE(#1,130,2,U,10,0,
-12,#2,130,16,U,7,0,-12):: !AERE
O E PARA'
390 V=INT(RND*9)+1 :: L=INT(RND*
4)+1 :: !FATTORI VENTO & PESO
400 DISPLAY AT(1,1):"PARA'RIMAST
I";PARA;"--SCORE";POINT
410 DISPLAY AT(24,2):"VENTO:";L*
2;"NODI -- PESO";(V*25)+50
420 CALL KEY(1,X,Y)
430 IF X=18 THEN CALL PATTERN(#2
,131)ELSE 420
440 CALL MOTION(#2,V,L):: CALL S
OUND(30,-6,5,150,5)
450 CALL COINC(#2,#7,Z,C1)
460 CALL COINC(#2,#8,Z,CC)
470 CALL COINC(#2,#9,Z,CCC)
480 IF (C=-1)+(CC=-1)+(CCC=-1)TH
EN 510
490 CALL COINC(#2,#3,50,R):: IF
R=-1 THEN 570
500 GOTO 450
510 CALL MOTION(#2,0,0):: CALL P
ATTERN(#2,46):: CALL SOUND(-1500
,5995,4)
520 FOR T=950 TO 1500 STEP 50 ::
CALL SOUND(50,T,31):: NEXT T
530 POINT=POINT-25*(C=-1)-50*(CC
=-1)-75*(CCC=-1)
540 CALL DELSPRITE(#1,#2):: DISP
LAY AT(13,5):"MISSIONE COMPIUTA"

550 FOR I=1 TO 150 :: NEXT I
560 CALL HCHAR(13,5,32,221):: GOT
O 370
570 CALL MOTION(#2,0,0):: CALL S
OUND(200,-4,3):: CALL PATTERN(#2
,43)
580 FOR I=1 TO 200 :: NEXT I ::
CALL PATTERN(#2,42)
590 CALL DELSPRITE(#1,#2):: DISP
LAY AT(13,3):"HAI CICCATO L'ATTE
RRAGGIO"

```

```

600 POINT=POINT-10 :: PARA=PARA-
1
610 FOR J=1 TO 150 :: NEXT J ::
CALL HCHAR(13,3,32,27)
620 GOTO 370
630 CALL HCHAR(1,1,32,29):: CALL
HCHAR(24,1,32,29)
640 FOR I=450 TO 850 STEP 25 ::
CALL SOUND(50,1,3):: NEXT I
650 FOR T=850 TO 450 STEP -25 ::
CALL SOUND(50,T,3):: NEXT T
660 DISPLAY AT(7,10):"GAME OVER"

```

```

670 DISPLAY AT(9,6):"HAI FATTO";
POINT;"PUNTI"
680 DISPLAY AT(12,2):"VUOI RITEN
TARE? (S/N)"
690 ACCEPT AT(12,25)VALIDATE("SN
"):RS$
700 IF RS$="N" THEN 730
710 CALL HCHAR(12,4,32,26):: CAL
L HCHAR(7,12,32,9):: CALL HCHAR(
9,6,32,24)
720 FL=1 :: GOTO 270
730 CALL CLEAR :: CALL DELSFRIE
(ALL):: CALL SCREEN(14):: DISPLA
Y AT(12,10):"CI VEDIAMO!"
740 GOSUB 920 :: END
750 CALL CLEAR :: CALL SCREEN(12
)
760 PRINT "LANCIA I TUOI PARACAD
UTISTI","SULLE PIAZZOLE DA 75, 5
0, 0"
770 PRINT "25 PUNTI. LANCIALI CO
N IL","TASTO DEL JOYSTICK #1 O C
ON","IL TASTO <Q>." :
780 PRINT "SE SBAGLI, CADRAI NELL
'OCEANO',"PERDENDO IL PARA'E IO
PUNTI."
790 PRINT "HAI A DISPOSIZIONE PE
R OGNI","PARTITA IO PARACADUTIST
I." : : : PRINT "LA VELOCITA' DE
L VENTO ED IL"

```

```

800 PRINT "PESO DI OGNI PARA' E'
","MOSTRATO A FONDO-SCHERMO:","T
IENI CONTO DI VELOCITA' DI"
810 PRINT "DISCESA E DI SPOSTAME
NTO --"
820 PRINT "PRIMA DI EFFETTUARE O
GNI","LANCIO, VALUTALE BENE!": :

```

```

830 PRINT TAB(10);"IN BOCCA AL L
UPO!": :
840 PRINT TAB(4);"PREMI UN TASTO
..."
850 CALL KEY(0,K,S):: IF S=0 THE
N 850
860 CALL CLEAR :: DISPLAY AT(8,6
):"ESPERIENZA DEI PARA'?"
870 DISPLAY AT(11,2):"<P>IVELLO
O <V>ETERANO"
880 ACCEPT AT(8,27)VALIDATE("VP"
):CS$
890 IF CS$="V" THEN 910
900 CALL MAGNIFY(2):: Z=10 :: GO
TO 140
910 Z=5 :: GOTO 140
920 CALL SOUND(300,330,3):: CALL

```

```

SOUND(300,392,3)
930 CALL SOUND(500,392,3):: CALL
SOUND(200,349,3)
940 CALL SOUND(100,330,3):: CALL
SOUND(200,294,3)
950 CALL SOUND(300,330,3):: CALL
SOUND(300,349,3)
960 CALL SOUND(300,370,3):: CALL
SOUND(300,392,3)
970 CALL SOUND(250,440,3):: CALL
SOUND(150,524,3)
980 CALL SOUND(500,524,3)
990 CALL SOUND(300,583,3):: CALL
SOUND(100,523,3)
1000 CALL SOUND(200,440,3):: CAL
L SOUND(300,392,3)
1010 RETURN

```

Corsa campestre

MSX

Questo è un gioco per persone veramente spor-
tive. Siete impegnati in una estenuante corsa
campestre sulla distanza di 1000 metri. Il per-

corso è reso impegnativo dalla presenza di pro-
fonde buche che voi dovete saltare premendo
la barra-spazio; agendo sui tasti cursore pote-

te aumentare o diminuire la velocità della corsa. Ma i problemi non terminano qui: infatti imperversa sulla zona un violento temporale, e voi dovrete essere abili ad evitare di essere colpiti dai fulmini che vi cadano vicini. La caduta in una buca vi farà perdere solo dei secondi preziosi, ma l'essere colpiti da una saetta vi farà naturalmente abbandonare la gara.

In questo programma viene utilizzata una routine in linguaggio macchina per lo scroll dello schermo. È quindi possibile, modificando la linea 180, il cui INTERVAL richiama la subroutine scroll, accelerare o rallentare lo scorrimento dello sfondo.

La linea 90, invece, controlla la velocità del corridore: provate a modificarne i valori.

```
1 REM CORSA CAMPESIRE
2 REM INIZIAZIONE
3 GOSUB80000
10 KEYOFF.COLOR 15,4,7:SCREEN1,3:FORI=1T
032 REMD A=A*CHR(0):NEXT:SPRITE(0)=
A*.A*""
15 FORI=1T032:READO:A=A*CHR(0):NEXT:S
PRITE(1)=A:A*""
20 FORI=1T032:READO:A=A*CHR(0):NEXT:S
PRITE(2)=A:A*""
25 FORI=1T032:READO:A=A*CHR(0):NEXT:S
PRITE(3)=A:A*""
30 FORI=60350T060401:READO:POKEI,0:NEX
T DEFUSK2=60350:POKE60346,0:POKE60347,
24:POKE60348,1.223:POKE60349,25
35 FORI=10T201215:READO:VPOKEI,0:NEXT
50 FORI=1T031:PRINT:NEXT.PRINTTAB(0)"COR
SA CAMPESIRE"
55 DEFUSR=60000:DEFUSR1=60110:POKE5999
91,1:POKE59996,10:POKE59997,6:POKE5999
8,3
60 PUTSPRITE0,(0,55),1
65 IFINKEY=""THENIFVPEEK(6914)=0THENVPO
KE6914,4ELSEVPOKE6914,8ELSEGOTO000
70 D=USR1(0):GOTO65
80 PRINT:PRINT:INPUT"Pressi il tasto di difficol
ta' (1-3)",L0
85 IFL0<1ORL0>3THEN80
90 POKE59999,1,2
99 REM DISEGNO SCHERMO
```

```
100 PUTSPRITE0,(100,200):CLS:FORI=6009T0
6879:VPOKEI,134:NEXT
105 VPOKE6592,134:VPOKE6593,134:VPOKE659
4,134:VPOKE6595,139:VPOKE6596,138:FORI=6
597T06606:VPOKEI,134:NEXT VPOKE6607,139:
VPOKE6608,138
110 VPOKE6560,135:VPOKE6561,135:VPOKE656
2,136:VPOKE6563,137:FORI=6565T06574:VPOK
EI,135:NEXT VPOKE6575,137:FORI=6578T0658
6:VPOKEI,135:NEXT:VPOKE6587,136:VPOKE658
8,137:VPOKE6589,138
115 VPOKE6590,135:VPOKE6591,135
120 VPOKE6152,142:FORI=6153T06163:VPOKEI
,140:NEXT:VPOKE6164,146:VPOKE6185,142:FO
RI=6186T06192:VPOKEI,140:NEXT:VPOKE6193,
144:VPOKE6194,144:VPOKE6195,145
125 VPOKE6209,150:FORI=6210T06215:VPOKEI
,149:NEXT:VPOKE6216,147:VPOKE6218,143:FO
RI=6219T06223:VPOKEI,140:NEXT:VPOKE6224,
146:VPOKE6233,150:VPOKE6234,149:VPOKE623
5,149:VPOKE6236,147
```

```
130 VPOKE6240,141:FORI=6241T06248:VPOKEI
,140:NEXT:VPOKE6249,148:VPOKE6250,147:VP
OKE6251,143:VPOKE6252,144:VPOKE6253,144:
VPOKE6254,144:VPOKE6255,145:VPOKE6259,15
0:FORI=6260T06263:VPOKEI,151:NEXT
135 VPOKE6264,151:FORI=6265T06268:VPOKEI
,140:NEXT:VPOKE6269,148
140 VPOKE6272,142:FORI=6273T06283:VPOKEI
,140:NEXT:VPOKE6284,148:VPOKE6290,141:FO
RI=6291T06301:VPOKEI,140:NEXT:VPOKE6302,
148
145 VPOKE6305,142:FORI=6306T06315:VPOKEI
,140:NEXT:VPOKE6316,146:VPOKE6322,141:FO
RI=6323T06327:VPOKEI,140:NEXT:VPOKE6328,
146:VPOKE6329,143:FORI=6330T06334:VPOKEI
,140:NEXT
150 VPOKE6338,143:FORI=6339T06346:VPOKEI
,144:NEXT:VPOKE6347,145:VPOKE6354,142:FO
RI=6355T06358:VPOKEI,140:NEXT:VPOKE6359,
146:VPOKE6362,143:VPOKE6363,144:VPOKE636
4,144:VPOKE6365,144:VPOKE6366,145
155 VPOKE6387,143:VPOKE6388,144:VPOKE638
9,144:VPOKE6398,145
170 PUTSPRITE0,(154,72)
180 INTERVALON:ONINTERVAL=15-2L6GOSUB80
0
185 SPRITEON:ONSPRITEGOSUB800
190 STRIG(0)ON:ONSTRIGGOSUB700:TIME=U
199 REM CONTROLLO
200 B=USR(D):K=INT((VPEEK(6913)+B)/0):IF
K=PC0THENNDT=DT+B*(K-PC)
205 PC=K:IFVPEEK(6560+PC)<135ANDJ=0ANDP
C<32THENGOSUB800
210 IFJ<0THENJ=J+1:IFJ=10-2L0THENJ=0:V
POKE6914,0
220 GOSUB1000
230 IFND(1)<0,07INENGOSUB300
240 IFLT<0THENGOSUB320
290 GOTO200
299 REM LAHPO
300 IFL1<01HENRETURN
305 PUTSPRITE3,(INT(RND(1)*255),50),11:L
T=1:PLAY"12+2008+10+45":RETURN
320 L1=LT+1
330 IFLT=5THENPUTSPRITE3,(100,200),11:L
T=U:RETURN
350 RETURN
499 REM SCROLL
500 D=USR2(0)
510 D1=DT+1
515 IFJ<0THENRETURN
520 STRIG(0):OFF:11=VPEEK(6914):IF11=4THE
N11=0ELSE11=4
```


Robot

Un esercito di robot vi sta attaccando e voi dovete difendervi utilizzando il cannone-laser. Alcuni robot, che scompaiono all'improvviso, vi permettono di guadagnare un maggior numero di punti. Usate il tasto 'I' per orientare il cannone e la barra-spazio per sparare. Note:

rete che la velocità del gioco aumenta con il passare dei secondi. Questo effetto è causato dalla linea 155: provate a modificare il valore della POKE 59999 e vedrete alterata tutta la velocità del gioco.

```
1 GOSUB60000
5 TIME=0
15 COLOR 15,4,7:SCREEN1,2:CLS:KEYOFF.PR1
NT" R O B O T ":PRINT:PRINT:PRIN
T:PRINT
20 FORI=1TO8:READQ:A$=A$+CHR$(Q):NEXT:SP
RITE$(Q)=A$:A$=""
25 FORI=1TO32:READO:A$=A$+CHR$(O):NEXT:S
PRITE$(1)=A$:SPRITE$(2)=A$:SPRITE$(3)=A$
:SPRITE$(4)=A$:A$=""
30 FORI=1TO32:READO:A$=A$+CHR$(O):NEXT:S
PRITE$(5)=A$:A$=""
35 FORI=1TO32:READO:A$=A$+CHR$(O):NEXT:S
PRITE$(6)=A$
40 DEFUSR=60120:
45 D(1)=2:D(2)=3:D(3)=8:D(4)=1
50 INPUT" SEI PRONTO ?":X$
60 P=1
99 REMSCHERNO
100 CLS:COLOR 11,6,6
105 FORI=6158TO6894STEP32:FORJ=8TO2:VPOK
EI+J,255:NEXT:NEXT
110 FORI=6464TO6559:VPOKEI,255:NEXT
115 FORI=6445TO6449:VPOKEI,20:NEXT:FORI=
6464TO6528STEP32:VPOKEI+13,20:VPOKEI+17,
20:NEXT
120 FORI=6573TO6577:VPOKEI,20:NEXT
125 PUTSPRITE5,(116,83),1:GOSUB1000
149 REM NUOVO GIRO
150 AB=0:R=R+1:PUTSPRITE1,(116,8),8:PUTS
PRITE2,(0,83),1:PUTSPRITE3,(116,192),8:P
UTSPRITE4,(255,83),1
155 POKES9999!,INT(R/3)+1
160 GOSUB350:SRIG(8)ON:ONSTRIG6GOSUB320
170 SPRITIEON:ONSPRITEGOSUB500
175 FORI=1TO4:S(I)=1:NEXT
199 REM CONTROLLO
200 IFF=1THEN220
210 GOSUB300
220 IFF=1THENGOSUB400
230 SPRITEOFF:GOSUB450:SPRITEON
235 ST$=SIR$(TIME):DG=VAL(RIGHT$(ST$,1))
240 IFDG>10RRND<(1)<.95THEN200
245 IFRND<(1)<.5THENPUTSPRITE2,(116,200):
S(2)=0:GOTO260
250 PUTSPRITE4,(208,200):S(4)=0
260 AB=1:FORI=1TO4:IFS(I)=1THENAB=0
265 NEXT:GOTO200
```

```
299 REM LETTURA TASSIERA
300 P$=INKEY$:IFP$<>"I"ANDP$<>"":THEN
RETURN
305 P=P+1:IFP=5THENP=1
310 GOSUB350:RETURN
320 IFF=1THENRETURN
322 PLWY"119*380*10*37"
325 F=1:ONP60TO330,335,340,345
330 PUTSPRITE0,(120,78),1:RETURN
335 PUTSPRITE0,(103,87),1:RETURN
340 PUTSPRITE0,(120,103),1:RETURN
345 PUTSPRITE0,(137,87),1:RETURN
349 REM DISEGNO BUNKER
350 VPOKE6447,20:VPOKE6509,20:VPOKE6575,
20:VPOKE6513,20
360 ONP60TO335,370,375,380
365 VPOKE6447,255:RETURN
370 VPOKE6509,255:RETURN
375 VPOKE6575,255:RETURN
380 VPOKE6513,255:RETURN
399 REM MOVIMENTO BOMBA
400 IF=PEEK(59999!):POKE59999!,8:POKE599
99!,2
405 POKE59998!,P-1:D=USR(D):POKE59999!,T
P
410 ONP60TO415,420,425,430
415 IFVPEEK(6912)<5THENF=0:PUTSPRITE0,(2
00,200):RETURN ELSE RETURN
420 IFVPEEK(6913)<5THENF=0:PUTSPRITE0,(2
00,200):RETURN ELSE RETURN
425 IFVPEEK(6912)>107THENF=0:PUTSPRITE0,
(200,200):RETURN ELSE RETURN
430 IFVPEEK(6913)>250THENF=0:PUTSPRITE0,
(200,200):RETURN ELSE RETURN
449 REM MOVIMENTO NEMICO
458 SM=SM+1:IFSM=5THENSN=1
455 IFS=1THEN150
460 IFS(SM)=0THEN450
465 POKE59999!,SM:POKE59998!,D(SM):D=USR
(D)
470 ONSM60TO475,480,485,490
475 IFVPEEK(6916)>62THEN600ELSERETURN
480 IFVPEEK(6921)>95THEN600ELSERETURN
485 IFVPEEK(6924)>111THEN600ELSERETURN
490 IFVPEEK(6929)>145THEN600ELSERETURN
500 STPIG(8)OFF:SPRITEOFF:F=0:IFP=10RP+3
THENSC=SC+20ELSESC=SC+100
```

```

510 GOTO1000.SCP)=0 PUTSPRITEP,(10,200)
VPKE6914,24
520 PLAY"t255164+64+70+10062+2"
530 FOR"=1T0150 NEXT:VPKE6914,0 PUTSPRI
TE0,(200,200)
535 AB=1:FOR"=1T04:IFS(1)=1THENAB=0
540 NEXT:STRIG(0)/ON SPRITEON:RETURN
599 REM GAME OVER
600 STRIG(0)OFF
630 CLS SCREEN1 PRINT" GAME OVER
".PRINT PRINT PRINT:PRINT:PRINT"MAI TOTA
LIZZATO ",SC,"PUNTI"
640 END
999 REM AGGIORNAMENTO PUNTEGGIO
1000 FOR"=1T023:PRINT:NEXT:PRINT"PUNTI";
SC,CHR$(11);:RETURN
9999 REM DATA SPRITES
10000 DATA24,24,60,255,255,60,24,24
10010 DATA167,48,96,96,224,193,226,254,2
31,199,195,224,96,96,48,16,8,12,6,6,7,13
1,71,127,231,227,195,7,3,3,6,4
10020 DATA1,1,1,3,15,12,27,251,251,27,12
,15,3,1,1,1,128,128,128,192,240,48,216,2
23,223,216,48,240,192,128,128,128
10030 DATA128,4,64,0,144,5,48,134,67,177
,68,16,68,0,128,16,4,0,129,16,6,129,146,
34,25,162,16,65,0,8,0,129
60000 CLS:RESTORE60020
60010 FOR"=60000:T060217:READO:POKE1,O:
NEXT.RESTORE:RETURN
60020 DATA219,170,230,240,33,92,234,94,2
46,8,0,0,0,0,0,0,0,0,0,211,170,219,1
69,71,6,0,0,0,0,0,0,0,0

```

```

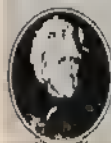
60030 DATA0,0,0,230,128,194,151,234,203,
31,202,151,234,62,3,58,94,234,195,209,23
4,128,230,64,194,170,234,203,83,202,170,
234,62,2
60040 DATA50,94,234,195,209,234,128,230,
32,194,189,234,203,67,202,189,234,62,0,5
0,94,234,195,209,234,120,230,16,194,208,
234,203,75,202,208,234,62,1,50,94,234,19
5,209,234,201,62,0,50,93,234,58
60050 DATA93,234,203,39,203,39,79,6,0,33
,0,27,4,58,95,234,71,58,94,234,87,254,0,
194,250,234,205,27,235,121,152,79,195,44
,235,254
60060 DATA1,194,0,235,35,205,27,235,195,
244,234,254,2,194,20,205,205,27,235,121,
128,79,195,44,235,35,205,27,235,195,14,2
35,125,243,211
60070 DATA153,124,211,153,0,0,0,0,219,15
2,79,183,251,201,125,243,211,153,124,246
,64,211,153,121,211,152,251,201

```

E' IN EDICOLA

Bit,
la prima rivista europea
di personal computer,
software, accessori,
la più prestigiosa
e più diffusa in Italia

con tutta la competenza del



**GRUPPO
EDITORIALE
JACKSON**

Tiro al bersaglio

Mettete a dura prova la vostra mira e la vostra prontezza di riflessi con questo divertente e molto ben congegnato programma. Ogni partita che effettuerete vi costerà 1 dollaro, fortunatamente immaginario, ma se sarete degli abili tiratori potrete rifarvi vincendo delle somme che variano a secondo dei bersagli colpiti, che sono rappresentati da dei poveri uccellini. Il bersaglio più basso, il più lento e quindi il più semplice da colpire, vale 20 centesimi, quello centrale, leggermente più veloce, 50 centesimi, quello più alto, il bersaglio pazzo, che si muove in maniera imprevedibile ed è quindi

molto difficile da mirare, vi farà vincere 1 dollaro. Per ogni partita avete a disposizione 5 colpi da utilizzare senza limiti di tempo, terminati i quali vi verrà chiesto se desiderate giocare ancora, nel qual caso vi viene addebitato il costo della nuova partita, altrimenti verrà visualizzata la vostra situazione. Per sparare è sufficiente premere la barra-spazio. Se volete modificare la velocità del movimento dei bersagli rendendo il gioco più facile oppure più difficile, potete agire sui valori della linea 350 per il bersaglio centrale e 410 per il bersaglio basso.

```
1 REM TIRO AL BERSAGLIO
5 GOSUB60000
10 REM INIZIALIZZAZIONE
15 SCREEN 2:COLOR 1,15,15:KEYOFF:PRINT
   TIRO AL BERSAGLIO
20 FOR I=1 TO 32:READ A$=A$+CHR$(Q):NEXT S
   PRITE$(0)=A$:A$=""
35 FOR I=1 TO 32:READ A$=A$+CHR$(Q):NEXT S
   PRITE$(1)=A$:A$=""
50 FOR I=1 TO 32:READ A$=A$+CHR$(Q):NEXT S
   PRITE$(2)=A$:A$=""
65 FOR I=1 TO 32:READ A$=A$+CHR$(Q):NEXT S
   PRITE$(3)=A$:A$=""
80 FOR I=1 TO 32:READ A$=A$+CHR$(Q):NEXT S
   PRITE$(4)=A$:A$=""
95 FOR I=1 TO 32:READ A$=A$+CHR$(Q):NEXT S
   PRITE$(5)=A$:A$=""
110 FOR I=1 TO 32:READ A$=A$+CHR$(Q):NEXT S
   PRITE$(6)=A$:A$=""
125 FOR I=1072 TO 119:READ VPOKE1,Q:NEXT
140 DEFUSR=6018:POKE5999!,2:POKE5999!,
   1:POKE5999!,5:PUTSPRITE2,(255,80),4
155 D=USR(D):IF INKEY$<>"" THEN 80
170 D=RND(1):IF VPEEK(6922)=8 THEN VPOKE6922
   12 ELSE VPOKE6922,8
185 FOR I=1 TO 120:NEXT:GOTO65
200 DH=1
99 REM DISLGO SCHERMO
100 CLS:VPOKE6922,8:PUTSPRITE2,(255,36),
   12:PUTSPRITE3,(140,36),12:PUTSPRITE0,(10
   0,16),8:PUTSPRITE4,(0,64),4:PUTSPRITE5,(
   115,64),4
110 FOR I=6145 TO 6206:VPOKE1,23:NEXT:FOR I=
   6729 TO 6742:VPOKE1,23:NEXT:FOR I=6793 TO 680
   6:VPOKE1,23:NEXT
115 FOR I=6816 TO 6847:VPOKE1,139:NEXT:FOR I=
   6848 TO 6874:STEP32:FOR J=0 TO 1:VPOKE1+J,134
   :VPOKE1+J+24,134:NEXT:NEXT
```

```
120 VPOKE6144,24:VPOKE6728,24:VPOKE6176,
   26:VPOKE6792,26:VPOKE6175,25:VPOKE6743,2
   5:VPOKE6207,27:VPOKE6807,27
125 VPOKE6760,22:VPOKE6775,22:VPOKE6783,
   135:VPOKE6735,136:VPOKE6767,137
130 START(0)=0:NSTR16GOSUB700
140 GOSUB800
155 REM CONTROLLO
200 GOSUB300:GOSUB350:GOSUB400
210 IF DH=0 THEN GOSUB450
220 IF DM=1 THEN GOSUB800:DM=0
250 GOTO200
299 REM MOVIMENTO BERSAGLIO SUP.
300 POKE5999!,7:POKE5999!,0:IF RND(1)<.
   5 THEN POKE5999!,1 ELSE POKE5999!,3
320 IF VPEEK(6913)>194 THEN POKE5999!,1 ELSE
   IF VPEEK(6913)<65 THEN POKE5999!,3
330 D=USR(D)
335 IF RND(1)<.5 THEN VPOKE6914,4 ELSE VPOKE
   6914,8
340 RETURN
349 REM MOVIMENTO BERSAGLIO MEDIO
350 POKE5999!,13:POKE5999!,2:K1=VPEEK(
   6922):VPOKE6922,VPEEK(6926):VPOKE6926,K1
360 POKE5999!,1:POKE5999!,3:D
   =USR(D)
370 RETURN
399 REM MOVIMENTO BERSAGLIO BASSO
400 C2=(C2+1) MOD 2:POKE5999!,4:IF C2=0 THEN
   HK1=VPEEK(6930):VPOKE6930,VPEEK(6934):VP
   OKE6934,K1
410 POKE5999!,4:POKE5999!,3:D=USR(D):P
   OKE5999!,5:D=USR(D)
430 RETURN
449 REM NUOVO BERSAGLIO
450 ON(DH+1)GOTO465,469,470,475,480,485
465 PUTSPRITE0,(100,16):GOTO490
470 VPOKE6922,8:VPOKE6926,12
472 GOTO490
```

```

475 VPOKE6922,0:VPOKE6926,12
477 GOT0490
480 VPOKE6930,16:VPOKE6934,20
483 GOT0490
485 VPOKE6930,16:VPOKE6934,20
490 DM=-1:RETURN
499 REM BERSAGLIO COLPITO
500 PLAY*150n1200n10n74r30n74r30n74r30n7
4r30n74r30n74r30n74*:00KDH+1*60T0505,508
,510,515,520,525
505 MM+1:VPOKE6914,24:FOR12=1T0500:NEXI
507 VPOKE6914,0:GOT0540
510 MM+.5:VPOKE6922,24:FOR12=1T0500:NEX
T
512 VPOKE6922,0:GOT0540
515 MM+.5:VPOKE6926,24:FOR12=1T0500:NEX
T
517 VPOKE6926,12:GOT0540
520 MM+.2:VPOKE6930,24:FOR12=1T0500:NEX
T
522 VPOKE6930,16:GOT0540
525 VPOKE6934,24:MM+.2:FOR12=1T0500:NEX
T
527 VPOKE6934,20
540 DM=1:STRIG(0):DN:RETURN
549 REM BERSAGLIO MANCATO
600 STRIG(0):DN VPOKE6191,138:FOR11=1T010
0:NEXT:VPOKE6191,23
620 DM=1:RETURN
699 REM SPARO
700 STRIG(0):OFF:PLAY*164n60n27*:NS=NS+
1
710 IFABS(VPEEK(6929)-114)<3:INENDH=4:GOT
0500
720 IFABS(VPEEK(6933)-114)<3:THENDM=5:GOT
0500
730 IFABS(VPEEK(6921)-114)<3:THENDM=2:GOT
0500
740 IFABS(VPEEK(6925)-114)<3:THENDM=3:GOT
0500
750 IFABS(VPEEK(6913)-114)<3:THENDM=0:GOT
0500
760 GOT0600
799 REM AGGIORNAMENTO VINCITA
800 IFM=>0:THENNS=" Vinci : $"ELSE$="
Perdi. $"
805 IFM<0:INENI3=INT(-M):I4=-M+13:ELSEI3
=INT(M):I4=M-13
807 FORI=1T022:PRINT:NEXT PRINI3:PRINT
USING"000",I3,:PRINTUSING"000",I4:PRINTC
HR$(11),
810 IFM>4:INEN05ELSEIREIURN
849 REM ALTRI COLPI?
850 PLAY*110n14n2000n30n40n20n27*:STRIG(
0):OFF:FORI=1T023:PRINI1:NEXT:PRINI1*Altri
5 colpi? 7 $(S) " :CHR$(11):
855 $*=INKEY$ IFX$=""THEN05ELSEIFX$<>"5
"ANDX$<>"6"ANDX$<>"H"ANDX$<>"n"THEN855
860 IFX$="N"ORX$="n"THEN900
865 M=M-1:NS=0:GOSUB800
870 FORT=6882T06910:VPOKEI,32:NEXT:STRIG
(0):DN:RETURN
899 REM GAME OVER
900 SCREEN1:IFM<0:THENPRINT"PERDI :$":P
RINTUSING"000.00":M:GOT0990
920 PRINT"MAI VINTO :$":PRINTUSING"0
00.00",M:GOT0990

```

```

990 IFINKEY$<>"*":THEN05ELSE990
999 REM DATA SPRITES
1000 DATA128,64,96,56,28,13,7,3,1,2,0,0,
0,0,0,0,1,2,6,20,56,176,224,192,128,64,0
,0,0,0,0,0
10010 DATA0,0,0,0,0,1,7,15,25,50,32,32,3
2,0,0,0,0,0,0,0,8,120,224,240,152,76,4,4
,4,0,0,0
10020 DATA0,0,0,0,3,14,24,12,6,115,255,6
3,7,0,0,0,0,0,0,0,252,32,64,32,240,255,2
55,254,252,248,0,0
10030 DATA0,0,0,0,1,115,255,63,3,7,7,3,0
,0,0,0,0,0,0,0,240,254,255,255,252,248,2
48,240,252,62,15,0
10040 DATA0,0,0,0,63,4,2,4,63,255,255,12
7,63,31,0,0,0,0,0,0,192,112,24,40,224,20
6,255,252,224,0,0,0
10050 DATA0,0,0,0,31,127,255,255,63,31,3
1,31,63,124,240,0,0,0,0,0,120,206,255,25
2,192,224,224,120,0,0,0,0,0
10060 DATA130,200,96,49,25,45,39,75,19,5
6,6,4,65,144,4,34,1,35,6,140,216,177,224
,201,196,144,66,32,4,18,120,1
10099 REM DATA CHARACTERI
10100 DATA170,85,170,85,170,85,170,85
10102 DATA96,32,32,32,32,32,32,32
10104 DATA32,32,48,48,40,40,112,56
10106 DATA48,56,56,56,60,60,60,60
10108 DATA0,0,24,36,02,133,64,18
10110 DATA170,85,170,85,0,0,0,0
60000 CLS:RESTORE60020
60010 FORI=60000:T060217*:REASQ:POKEI,0:
NEXT RESTORE:RETURN
60020 DATA219,170,230,240,33,92,234,94,2
46,8,0,0,0,0,0,0,0,0,0,0,0,211,170,219,1
69,71,0,0,0,0,0,0,0,0
60030 DATA0,0,0,230,128,194,151,234,203,
91,202,151,234,62,3,50,94,234,195,209,23
4,120,230,64,194,170,234,203,63,202,170,
234,62,2
60040 DATA50,94,234,195,209,234,120,230,
32,194,189,234,203,67,202,189,234,62,0,5
0,94,234,195,209,234,120,230,16,194,208,
234,203,75,202,200,234,62,1,50,94,234,19
5,209,234,201,62,0,50,93,234,58
60050 DATA93,234,203,39,203,39,79,6,0,33
,0,27,9,58,95,234,71,58,94,234,87,254,0,
194,250,234,205,27,235,121,152,79,195,44
,235,254
60060 DATA1,194,6,235,35,205,27,235,195,
244,234,254,2,194,20,235,205,27,235,121,
120,79,195,44,235,35,205,27,235,195,14,2
35,125,243,211
60070 DATA153,124,211,153,0,0,0,0,219,15
2,79,103,251,201,125,243,211,153,124,246
,64,211,153,121,211,152,251,201

```

FAI CRESCERE



**IL TUO TI99/4A CON LA
NUOVA FORMIDABILE
OFFERTA J.SOFT**



Dopo il successo ottenuto dalla prima proposta, ecco a disposizione del tuo TI-99/4A un assortimento di altri splendidi programmi, tra i quali il richiestissimo **EXTENDED BASIC**; il **MODULO ADVENTURE**, che ti permetterà di vivere appassionanti avventure e l'**EDITOR/ASSEMBLER**, indispensabile per il definitivo salto di qualità del tuo computer.

Ordina subito, per essere sicuro di ricevere i programmi al più presto.

EXTENDED BASIC

Cod. DTXNX01 - L. 270.000

Finalmente disponibile il linguaggio di programmazione compatibile con il BASIC TI-99/4A, con le seguenti aggiunte: righe multidichiarazioni, dichiarazioni IF-THEN-ELSE, accesso diretto allo schermo, sottoprogrammi BASIC con variabili locali, autoavviamento dei programmi su disco, facile controllo di fino a 28 sprite con concatenamento programmi BASIC, fusione codici, programmi protetti, dichiarazioni ON ERROR (Errore in atto) e funzioni logiche di Boole (AND, OR, NOT, IE, O, NO, ecc.).

Permette, con l'espansione della memoria, di programmare in linguaggio macchina.

MODULO AVVENTURA

Cod. DTXRX01 - L. 70.000

Si tratta del modulo di comando, da inserire nella console, che funge da interprete del linguaggio delle cassette sottodescritte.

Uno strumento indispensabile per arricchire la tua software-teca di nuove entusiasmanti avventure. Il modulo viene fornito con l'avventura "IL PIRATA".

Configurazione richiesta: registratore a cassetta.

Elenco delle "Avventure" che girano esclusivamente col MODULO AVVENTURA

Nove programmi entusiasmanti, semplici da usare ma difficili da finire.

All'inizio di ogni avventura viene proposta la situazione in cui ci si trova e viene posta la domanda: COSA DEVO FARE? Utilizzando gli oltre 200 vocaboli previsti dal vocabolario del computer, darai gli ordini conseguenti, per trovarti in situazioni sempre diverse fino a realizzare lo scopo del gioco, ogni volta differente per ogni titolo.

Avventure che stimolano la tua capacità di ragionare in modo logico e che richiedono ore, e anche settimane, per concludersi.

Voodoo Castle (Castello del Voodoo)

Cod. DTXMX32 - L. 30.000

Sei all'interno di una cappella di fronte ad una bara chiusa. Esplorando il castello, cercherai di raccogliere le informazioni necessarie per liberare il Conte Cristo dalla terribile maledizione che i suoi nemici gli hanno fatto.

Adventureland

Cod. DTXMX01 - L. 30.000

Ti trovi all'inizio della foresta di un mondo di fiaba. Esplorandolo, potrai scoprire 13 diversi tesori come pure il luogo particolare dove riporli.

Strange Odyssey (Strana Odissea)

Cod. DTXMX24 - L. 30.000

Inizia quando ti rendi conto di essere "naufragato" su un minuscolo pianeta e quindi devi riparare l'astronave per poter tornare a casa.

Mentre cerchi i pezzi di ricambio sul pianeta, tenterai di scoprire i segreti di un'antica civiltà, raccogliendo i tesori in cui ti imbattevi lungo la strada.

The Golden Voyage

Cod. DTXMX10 - L. 30.000

Comincerai con l'individuare il palazzo reale nella città persiana. Nel palazzo incontrerai l'anziano re al quale rimangono solo tre giorni di vita, a meno che non riuscirai a ringiovanirlo.

Portando soltanto una borsa d'oro, salperai in mare alla ricerca della mitica fontana della giovinezza.

Savage Island Series (Serie dell'Isola Selvaggia)

Cod. DTXMX35 - L. 30.000

L'avventura si divide in due parti ed inizia al margine di una giungla impenetrabile. Mentre esplori l'isola, incontrerai delle creature insolite. Dopo aver completato la prima parte, riceverai la parola d'ordine per iniziare la seconda parte, dove o diventerai il più grande eroe al mondo o andrai incontro a una fine rapida ed orribile.

Mystery Fox House (La casa del Mistero)

Cod. DTXMX17 - L. 30.000

Prima di procedere all'esplorazione, devi meditare su come fare per entrare nella Casa del Mistero. Una volta dentro, potrai vedere tutti gli angoli della casa, dove si nasconde un prezioso premio.

The Count (Il conte)

Cod. DTXMX26 - L. 30.000

Ti svegli da un pisolino e ti trovi in un letto con un paletto da tenda in mano. Sta a te scoprire chi sei, cosa stai facendo in Transilvania e perché il postino ha consegnato un flacone di sangue!

Pyramid of Doom (La Piramide del Destino)

Cod. DTXMX22 - L. 30.000

L'avventura inizia in un deserto, nei pressi di uno stagno dal quale fuoriesce un palo. Grazie alle tue ricerche, troverai una piramide solo di recente scoperta dalle sabbie mobili. Cercherai l'ingresso, raccoglierai tesori e cercherai di fuggire dalla piramide.

Ghost Town (La Città Fantasma)

Cod. DTXMX09 - L. 30.000

Tutte le città fantasma sono misteriose, ma questa ha un fascino particolare: è abitata da fantasmi veri ed è piena di tesori.

Esplora tutti i vecchi edifici per vedere quanti ne riuscirai a trovare.

Music Maker (Il compositore di Musica)
Cod. DTXQX01 - L. 42.000 - con istruzioni in italiano

Un ottimo programma sia per chi vuole avvicinarsi alla musica senza cognizioni di base, sia per chi deve studiare composizione direttamente sul programma e sia per chi utilizza il suono dal punto di vista tecnico ed ha interesse per tipo di frequenza, ecc.. Il programma prevede due opzioni: scrittura di partiture sul pentagramma: inserendo battuta per battuta è possibile scrivere partiture composte anche da tre voci e persino stamparne gli spartiti (in questo caso è necessaria una stampante e l'interfaccia RS232 o similare). L'altra opzione prevede la realizzazione delle melodie prendendo in esame gli Hertz di una singola nota.

Parsec
Cod. DTXMX21 - L. 42.000

PARSEC è la tua astronave, con la quale combatterai gli alieni ribelli e le loro navicelle, buttandole fuori rotta e emetterai un fuoco dal tuo laser che fa inaridire. Tenterai di sopravvivere alla "cintura" di asteroidi mortali.

Per attivare le capacità sonore di questa cartuccia, puoi utilizzare lo Speech Synthesizer.

Soccer (Gioco del Calcio)
Cod. DTXMX23 - L. 37.000

Inutile spendere troppe parole per questo apprezzatissimo programma. Cinque giocatori per squadra, con passaggi, tiri, intercettamenti, parate, cariche e molte altre tattiche del calcio. Potete persino avere il replay immediato di una rete, al rallentatore.

Ti Invaders (Invasori Ti)
Cod. DTXMX27 - L. 30.000

Il mondo viene attaccato da numerose creature odiose provenienti dallo spazio. Devi fare affidamento sulla tua agilità mentale e fisica per distruggere le creature multicolori con i missili a disposizione.

Car Wars (Guerre di Auto)
Cod. DTXMX06 - L. 30.000

Se ti emozionano le corse automobilistiche, abbinate alla possibilità di battere l'avversario con astute manovre, ecco il programma che fa per te. Cimenta la tua velocità ed abilità contro il computer mentre cerchi di avanzare sul circuito, evitando incidenti che farebbe saltare il tuo bolide fuori pista.

E ora un'altra eccezionale proposta:

Game Writer's Pack 1 & 2
Cod. DTXIX13 - L. 35.000
(prezzo per le 2 cassette)

Un'ottima realizzazione didattica divisa in due parti autonome ma complementari che insegna i principi (e li approfondisce) della programmazione BASIC, tramite la programmazione di videogiochi. Molto ben strutturata e comprensibile, utilizza un linguaggio semplice ed elementare. In ogni confezione è compresa una cassetta con i giochi di cui si parla nel corso della dissertazione. In totale 6 giochi per confezione.

La trattazione è basata sul TI-BASIC anche se non disdegna l'EXTENDED BASIC.

Fun - Pac (1-2-3)
Cod. DTXMX36 - L. 27.000
(prezzo per 3 cassette)

Tre cassette (non acquistabili separatamente), con 8 giochi complessivi ben strutturati ed avvincenti. Dalla corsa dei cavalli al gioco del golf. Dal salvataggio di scienziati dalla contaminazione nucleare alla navicella Supernova, un ottimo gioco "adventure".

E INFINE...

Editor/Assembler
Cod. DTXAX01 - L. 180.000

L'eccezionale programma che ti permette di programmare in linguaggio di programmazione TSM 9900 è dà accesso diretto a tutte le caratteristiche del sistema tra cui audio, voce, grafica ed I/O (Entrata/Uscita), oltre a mettere a disposizione la massima rapidità grazie al microprocessore a 16 bit del computer. Le routine in linguaggio di programmazione possono essere svolte come programmi in-

dipendenti oppure collegate a programmi TI BASIC o TI BASIC ESTESO con l'impiego di una chiamata di sottoroutine. Oltre al modulo SSS, il programma comprende due floppy-disk (uno dei quali contiene il gioco TOMBSTONE CITY) e il Manuale Utente che dà tutte le informazioni sulla composizione del software. L'EDITOR/ASSEMBLER richiede i seguenti optional:

— **Peripheral Expansion Box**
Cod. ETXOX01 - L. 320.000

— **32 K Expansion Card**
Cod. ETXOX02 - L. 340.000

— **Drive Control Card**
Cod. ETXOX03 - L. 440.000

— **Floppy disk drive Box**
Cod. ETXOX04 - L. 480.000

N. B. È possibile acquistare insieme gli optional ETXOX01 - ETXOX02 - ETXOX03 - ETXOX04, al prezzo complessivo di L. 1.250.000 anziché L. 1.580.000. In questo caso, il codice da indicare per l'acquisto dei 4 pezzi è ETXOX05

e, infine, freschi di stampa, esconati del 10% sul prezzo di copertina:

Paper Book - TI-99/4A (con cassetta) -
editore J.soft
Cod. ASOC007 - L. 16.200

— **35 Programmi per il TI-99/A4** -
editore MUZZIO
Cod. AMUC024 - L. 9.000

Una selezione dei migliori programmi tratti dal nostro settimanale PAPER SOFT, scelti accuratamente sulla base della loro originalità e fantasia.

Programmi divertenti e didattici, dal labirinto al tennis, dagli anagrammi alle conversioni metriche e tanti altri ancora.

RITAGLIA ED

INVIA IL TAGLIANDO A:

Spett. J.soft - Viale Restelli, 5 - 20124 Milano (MI)
Tel. 02/6888228-683797-6880841/2/3/

Ordino il seguente software/libri per il TI-99/4A:

cod.	cod.	cod.
cod.	cod.	cod.
cod.	cod.	cod.

per un totale di L. + L. 2.500 per contributo fisso di spese di spedizione.

Scelgo la seguente modalità di pagamento:

- ☐ pagherò in contrassegno al postino
- ☐ allego assegno (o contanti)
- ☐ verso l'imporlo sul C.C.P. n. 19445204
Intestato a J.soft (allego ricevuta)

Nome

Cognome

Via N.

CAP Città Prov.